AUTOMATIC ERROR ANALYSIS IN FINITE DIGITAL COMPUTATIONS, USING RANGE ARITHMETIC

VICTOR JOSEPH MONTE LEON

AUTOMATIC ERROR ANALYSIS IN FINITE DIGITAL COMPUTATIONS, USING RANGE ARITHMETIC

by

Victor Joseph Monte Leon
Lieutenant, United States Navy
B. S., Polytechnic Institute of Brooklyn, 1960

[redacted]

Submitted in partial fulfillment
for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

UNITED STATES NAVAL POSTGRADUATE SCHOOL

May 1966

ABSTRACT

The nature of generated machine error in finite digital calculations is discussed. The arithmetic of range numbers is developed, and examples are given demonstrating the use of range arithmetic as a tool for automatic error analysis. A computer program is developed, utilizing the TYPE OTHER feature of FORTRAN-63 in conjunction with the CDC-1604 digital computer, which enables the user to perform automatic error analysis during computation, and a number of programs are presented using this feature.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

1. Introduction.

The utilization of high-speed digital computers to perform lengthy and complex mathematical computations is widespread. Unfortunately, the user often assumes that the computer output will necessarily be accurate if his input data is accurate and his program logic is sound. This need not be the case, and such an assumption may well lead the user to incorrect or highly inaccurate results. The difficulty arises because of the very nature of digital computation.

A digital computer has a finite memory, and cannot carry within it infinite precision numbers, or perform infinitely accurate calculations. Each number stored in a digital computer must, of necessity, contain a finite number of digits. As a result, during each step of computation, the computer automatically truncates or rounds off numbers to conform with its memory storage requirements. Each arithmetic operation introduces an error in precision due to such truncation, which may accumulate with other such errors to produce sizeable inaccuracies in results. It is also possible that the errors may cancel and give a fairly accurate result; but, in either case, the user will have no knowledge of whether or not any error had accumulated in computation.

It would be desirable to eliminate precision errors entirely, but hardware limitations make it impossible to do so. It is, however, possible to estimate and control the errors arising in finite digital computations, by appropriate choice of an algorithm. One method is called range arithmetic, or interval arithmetic [1, 2]. This method enables the generated error to be analysed automatically within the computer, during computation.

The technique of range arithmetic replaces any real number of infinite precision by two numbers of finite precision, one of which is a lowerbound of the real number, and the other is a corresponding upperbound of the real number. We can then be certain that the real number lies in the range of the two finite precision numbers.

In this thesis, we will discuss the arithmetic of range numbers, the application of range arithmetic to digital computation, and present a computer program designed for use on the CDC-1604 computer in conjunction with FORTRAN-63, which will provide a means of automatic error analysis in digital computation.

2. The Arithmetic of Range Numbers.

In this section, we will discuss the properties of range numbers, and the arithmetic operations pertaining to them.

Let x be any real number. Associated with this real number is a range number X, which has the following properties:

(1) X is a closed bounded interval;

(2) X contains x.

Then define the range number X, associated with the real number x, to be the closed bounded interval

$$X = [xL, xU],$$

where xL and xU are defined by the transformations L and U as follows:

$$x \xrightarrow{L} xL$$

$$x \xrightarrow{U} xU$$

We may then write

$$x \xrightarrow{L,\ U} X = [xL, xU] \quad .$$

The transformations L and U are determined by the precision allowed in computation. If, for example, we would like to find the range number associated with e = 2.71828 ..... , and we are allowed only two significant figures, then

$$X = [2.7, 2.8] \ .$$

On the other hand, if allowed five significant figures, then

$$X = [2.7182, 2.7183] \ .$$

For integers (known with exact precision), the range number will have the same upper and lower bound, equal to the integer itself. Then, if $x = 3, \quad X = [3, 3]$.

We can see, then, that the range number X will represent the smallest closed interval containing x, for a given set of transformations L and U.

The arithmetic operations associated with range numbers can be defined in the following manner. Let (op) represent an arithmetic operation, and consider two range numbers

$$X = [xL, xU] ,$$

and

$$Y = [yL, yU] .$$

Then,

$$Z = X\,(op)\,Y = [zL, zU] ,$$

where

$$zL = \min[xL(op)yL, xL(op)yU, xU(op)yL, xU(op)yU]$$

$$zU = \max[xL(op)yL, xL(op)yU, xU(op)yL, xU(op)yU] .$$

It can be seen that the real number $z = x\,(op)\,y$ will be in the interval Z, and that Z represents the smallest interval that contains z.

Now let

(+) - denote range addition

(-) - denote range subtraction

(*) - denote range multiplication

(/) - denote range division

From the definition above, it can be seen that

(1) $Z = X (+) Y = [xL + yL, xU + yU]$

(2) $-X = [-xU, -xL]$

(3) $Z = X (-) Y = X (+) (-Y) = [xL - yU, xU - yL]$

(4) $X = Y$ if and only if

$xL = yL$, and $xU = yU$

(5) $Z = X (*) Y$, where

$zL = \min [xL * yL, xL * yU, xU * yL, xU * yU]$

$zU = \max [xL * yL, xL * yU, xU * yL, xU * yU]$

The values of zL and zU can also be obtained using a table of sign discriminations.

Using the table on page 14, we need only calculate one product for each end point, except in sign discrimination 5.

(6) $Z = X (/) Y$, where

$zL = \min [xL / yL, xL / yU, xU / yL, xU / yU]$

$zU = \max [xL / yL, xL / yU, xU / yL, xU / yU]$

Note, that in the case in which Y brackets zero, i. e., $Y = [-1, 3]$, range division is undefined. This is equivalent to forbidden division by zero in real arithmetic.

	xL	xU	yL	yU	Values of zL and zU
1	+	+	+	+	zL = xL*yL , zU = xU*yU
2	+	+	-	+	zL = xU*yL , zU = xU*yU
3	+	+	-	-	zL = xU*yL , zU = xL*yU
4	-	+	+	+	zL = xL*yU , zU = xU*yU
5	-	+	-	+	zL = min [xU*yL , xL*yU], zU = max [xL*yL , xU*yU]
6	-	+	-	-	zL = xU*yL , zU = xL*yL
7	-	-	+	+	zL = xL*yU , zU = xU*yL
8	-	-	-	+	zL = xL*yU , zU = xL*yL
9	-	-	-	-	zL = xU*yU , zU = xL*yL

TABLE OF SIGN DISCRIMINATIONS

FIGURE 1

We now will give some illustrative examples of computations using range arithmetic.

$$[a, b] (+) [c, c] = [a+c, b+c]$$

$$[a, b] (*) [c, c] = [ca, cb] \qquad \text{for } c \geq 0$$

$$[a, b] (*) [c, c] = [cb, ca] \qquad \text{for } c \leq 0$$

$$[a, b] (-) [a, b] = [a-b, b-a]$$

$$[a, b] (/) [a, b] = [a/b, b/a] \qquad \text{for } a > 0$$

$$[1, 2] (*) [-4, -3] = [-8, -3]$$

3. Range Arithmetic and Automatic Error Analysis.

In finite digital computation, errors are generated in the process of round-off, truncation, and normalization. One has no method of determining the seriousness of such errors unless some sort of error analysis is performed. A method is necessary that can evaluate the error generated at each step in the computation, and also keep track of the accumulated error. The techniques of range arithmetic are suited to this purpose. If, in addition to doing real arithmetic, we parallel the computation with range arithmetic, the range numbers calculated will keep track of the error, and the final result will be the smallest closed interval containing the "true", infinite precision result.

Error analysis via range arithmetic not only furnishes information as to the size of machine-generated errors, but it also can be utilized to evaluate the accuracy of alternative computational schemes. Indeed, the error generated in one type of computation may be quite different from that obtained using another method of computation. A few examples will illustrate this point.

Example 1.

Consider the problem of finding the roots of the equation

$$x^2 + 100,000x + 100 = 0 .$$

We will concern ourselves with finding the larger root of this equation, and we will assume that the computer can carry six significant figures. We will first calculate the result using ordinary arithmetic and the quadratic formula, and then use range arithmetic in conjunction with the quadratic formula. The notation will be that which is equivalent to normalized floating-point operations on a computer. For example, the number 100,000 will be denoted by .100000E6, where E6 denotes the power of ten multiplying .100000.

The equation

$$ax^2 + bx + c = 0$$

has, as its larger root,

$$x = (-b + \sqrt{b^2 - 4ac})/2a .$$

We then calculate our solution:

$$x = \frac{-.100000E6 + \sqrt{.100000E11 - .400000E3}}{.2E1}$$

$$= \frac{-.100000E6 + \sqrt{.100000E11 - .000000E11}}{.2E1}$$

$$= \frac{-.100000E6 + .100000E6}{.2E1}$$

$$= .000000E00$$

or

$$x = 0$$

Now, in order to do the same problem in range arithmetic, we assume some uncertainty in the coefficients of our original equation. We then replace the original coefficients by range numbers reflecting the uncertainty. For example, we will replace 100,000 by [99999.9, 100,001].

The equation to be solved now is

$$[.999999E0,\ .100001E1]\,x^2 + [.999999E5,\ .100001E6]\,x$$

$$+ [.999999E2,\ .100001E3] = 0\ .$$

Now,

$$b^2 = [.999999E5,\ .100001E6]^2$$

$$= [.99999E10,\ .100003E11]$$

$$ac = [.999999E0,\ .100001E1] \times [.999999E2,\ .100001E3]$$

$$= [.999998E2,\ .100003E3]$$

$$4ac = [.399999E3,\ .400012E3]$$

$$b^2 - 4ac = [.999998E10, .100003E11] - [.399999E3,\ .400012E3]$$

$$= [.999997E10, .100003E11]$$

$$\sqrt{b^2 - 4ac} = [.999998E5,\ .100002E6]$$

$$-b + \sqrt{b^2 - 4ac} = [.999998E5,\ .100002E6] - [.999999E5,\ .100001E6]$$

$$= [-.120000E4,\ .210000E4]$$

and

$$x = \frac{[-.120000E4,\ .210000E4]}{[.399999E3,\ .400012E3]}$$

$$= [-.299994E1,\ .525002E1]$$

By using range arithmetic, we have kept accurate error bounds and have found that our root is in the interval [- 2.9, 5.3], or [- .299994E1, .525002E1] . Our result tells us that the true value could be anywhere within the calculated interval. However, the interval is wide, and could contain the value zero, which was our answer using real arithmetic. We would like to tighten the interval, so that we may be more confident of the computer results. The next example will illustrate how, with a little knowledge of sources of error generated in finite calculations, we can obtain more satisfactory results.

Example 2.

Consider the formula for the larger root of a quadratic equation,

$$x = (- b + \sqrt{b^2 - 4ac}) / 2a$$

and divide numerator and denominator by b so that

$$x = (- 1 + \sqrt{1 - s}) b / 2a$$

where

$$s = 4ac / b^2 \quad .$$

Expanding the radical in a power series, we obtain

$$\sqrt{1 - s} = 1 - s/2 - s^2/8 - s^3/16 - \ldots\ldots\ldots$$

The equation for x becomes

$$x = - (s/2 + s^2/8 + s^3/16 + \ldots\ldots) b/2a \quad .$$

Now, for the equation

$$x^2 + 100,000x + 100 = 0$$

we get

$$s = \frac{(.400000E1)\,(.100000E1)\,(.100000E3)}{(.100000E6)^2}$$

$$= .400000E-7$$

$$s/2 = .200000E-7$$

$$s^2/8 = .200000E\text{-}15$$

$$b^2/2a = (.100000E6)/(.200000E1)(.100000E1)$$

$$= .500000E5$$

Then

$$x = -(.500000E5)(.200000E\text{-}7 + .200000E\text{-}15)$$

$$= -(.500000E5)(.200000E\text{-}7)$$

$$= -.100000E\text{-}2$$

or,

$$x = -.001$$

We now solve the same problem using range arithmetic, with the modified quadratic formula.

Our equation is

$$[.999999E0,\ .100001E1]\,x^2 + [.999999E5,\ .100001E6]\,x + [.999999E2,\ .100001E3] = 0 .$$

In example 2, we found that

$$b^2 = [.999998E10,\ .100003E11]$$

$$4ac = [.399999E3,\ .400012E3]$$

therefore,

$$s = \frac{[.399999E3,\ .400012E3]}{[.999998E10,\ .100003E11]}$$

$$= [.399987E\text{-}7,\ .400013E\text{-}7]$$

$$s/2 = [.399987E\text{-}7,\ .400013E\text{-}7]/[.200000E1,\ .200000E1]$$

$$= [.199994E\text{-}7,\ .200007E\text{-}7]$$

$$s^2 = [.159987E-14, \ .160009E-14]$$

$$s^2/8 = [.199983E-15, \ .200012E-15]$$

$$s/2 + s^2/8 = [.199994E-7, \ .200008E-7]$$

$$b/a = [.999999E5, \ .100001E6]/[.999999E0, \ .100001E1]$$

$$= [.999989E5, \ .100002E6]$$

$$b/2a = [.499994E5, \ .500005E5]$$

and

$$x = -[.499994E5, \ .500005E5]\,[.199994E-7, \ .200008E-7]$$

$$= [-.100005E-2, \ -.999958E-3]$$

or,

$$x = [-.00100005, \ -.000999958] \ .$$

Thus, we can see that the method in example 2 generates far less error than the straight application of the quadratic formula. Indeed, the previous examples illustrate the use of range arithmetic in evaluating a given computational technique.

4. QRANGE7 - Program Description and Usage.

The TYPE OTHER capability in the FORTRAN-63 compiler for the CDC-1604 computer allows the relatively simple implementation of range arithmetic for the user. Using the QRANGE7 package, the programmer need only declare all floating-point variables to be TYPE RANGE7(3), and follow the usual rules of FORTRAN programming. The QRANGE7 package supplies the necessary subroutines for range arithmetic computations.

Range arithmetic calculations are made carrying a triple of numbers (xF, xL, xU), rather than a double as stated previously. The first number in the triple is the ordinary floating-point result, as calculated using standard floating-point arithmetic. The remaining numbers in the triple are the lower and upper range numbers, respectively. Note that $xL \le xF \le xU$. In any range calculation, the "real" part of the triple is computed last and remains in the accumulator, so that transfers and comparisons (such as the IF statement) will follow the same logical branches as in ordinary floating-point arithmetic.

Standard floating-point arithmetic cannot be used on either lower or upper range numbers since, in floating-point calculations, round-off and normalization are done automatically, and would reduce the accuracy of the range interval. In QRANGE7, the arithmetic performed on range numbers is un-normalized, and separate subroutines are provided which truncate the lower range numbers and round up by one bit in the least significant position, or truncate, the upper range numbers as required. The un-normalized arithmetic is performed by unpacking the operands, putting them in un-normalized form, doing the appropriate arithmetic, re-normalizing and repacking the result.

Input/output may be accomplished in one of two ways. The programmer may use EQUIVALENCE statements to transfer input / output of range numbers as real numbers, where each range number is equivalent to three real numbers, or he may use the Input / Output for Multi-Word TYPE OTHER package [3] and transfer the range numbers directly in and out of the computer memory. The following examples will illustrate the use of these methods, and the use of QRANGE7.

A.

```
      TYPE RANGE7(3)  A7, B7, X7
      DIMENSION  A7(3), AR(9), B7(3), BR(9), X7(3, 3), XR(27)
      EQUIVALENCE  (A7, AR), (B7, BR), (X7, XR)
      READ  10, AR, BR
10    FORMAT(8F10.0)
      DO  20  I=1, 3
      DO  20  J=1, 3
20    X7(J, I) = A7(I)*B7(J)
      DO  30  I=1, 25, 3
30    PRINT  40, XR(I), XR(I+1), XR(I+2), XR(I), XR(I+1),  XR(I+2)
40    FORMAT(//15X, 3(E20.10, 10X)/15X, 3(020, 10X))
      END
```

In this example, we have assumed that the programmer reads in the three parts of the range numbers using real floating-point numbers that are simultaneously stored as the three parts of the range numbers by use of the EQUIVALENCE statement. The next example will illustrate the same program, but using the INPUT / OUTPUT for MULTI-WORD TYPE OTHER package [3].

B.

```
      TYPE  RANGE7(3)  A7, B7, X7
      DIMENSION  A7(3), B7(3), X7(3, 3)
      CALL  READ7
      READ  10, A7, B7
   10 FORMAT(8F10.0)
      DO  20  I=1, 3
      DO  20  J=1, 3
      X7(J, I) = A7(I)*B7(J)
      CALL  PRINT7
   20 PRINT  30,  X7(J, I)
   30 FORMAT(//15X, 3(E20.10, 10X) / 15X, 3(020, 10X))
      END
```

5. Matrix Inversion Using Range Arithmetic.

In order to test QRANGE7, and evaluate the errors generated in a particular program, a number of matrix inversions were run. In one case, the matrix to be inverted was generated randomly, using the RANF(-1) library function of FORTRAN-63. About 15 programs were run in this manner. In the second case, the input matrix was specified, and the individual elements of the matrix were assumed to be inaccurate. Thus, they were inputed as range numbers. Both cases used the MATINV2 subroutine provided by the U. S. Naval Postgraduate School computer facility, in conjunction with QRANGE7. Two range subroutines had to be used: ABS7, which takes the absolute value of a range number; and Q0Q06700, which complements the range accumulator [5]. (A listing of these subroutines appears in Appendix II.) The coding for both cases was as follows:

Case 1. Randomly Generated Matrices

```
      PROGRAM RANMAT
      TYPE RANGE7(3) XMAT,XINV
      DIMENSION XMAT(5,5),XINV(5,5),XJUNK(2000)
      READ 2000,N
      DO 90 I=1,N
   90 XJUNK(I) = RANF(-1)
      DO 10 I=1,5
```

```
      DO 10 J=1,5
      B(I,J) = RANF(-1)*1000
   10 XMAT(I,J) = B(I,J)
      CALL PRINT7
      DO 20 J=1,5
      DO 20 I=1,5
   20 PRINT 2001,XMAT(I,J),XMAT(I,J)
      CALL MATINV2(XMAT,XINV,5,5)
      CALL PRINT7
      DO 30 J=1,5
      DO 30 I=1,5
   30 PRINT 2001,XINV(I,J),XINV(I,J)
 2000 FORMAT(F10.0)
 2001 FORMAT(//15X,3(E20.10,10X)/15X,3(020,10X))
      END
```

Case 2. Input Matrix Containing Range Elements

```
      PROGRAM MMATRIX
      TYPE RANGE7(3) A,B
      DIMENSION A(3,3),B(3,3)
      CALL READ7
      DO 10 J=1,3
      DO 10 I=1,3
   10 READ 2000,A(I,J)
```

```
      CALL  PRINT7
      DO  20  J=1,3
      DO  20  I=1,3
20    PRINT  2001 , A(I, J) , A(I, J)
      CALL  MATINV2(A, B, 3, 3)
      CALL PRINT7
      DO  30  J=1,3
      DO  30  I=1,3
30    PRINT  2001 , B(I, J) , B(I, J)
2000  FORMAT(3F10.0)
2001  FORMAT(//15X , 3(E20.10, 10X) / 15X, 3(020, 10X))
      END
```

The results obtained using random matrices were highly accurate. For 5 x 5 matrices, seven decimal places of accuracy were obtained, and the range of error was small. It might be noted that the conversion of the real random matrix to a matrix of range numbers introduced no errors, and hence the resulting range intervals on the inverse elements were solely a consequence of digital computation. On the other hand, the error generated in the second case was quite large, and decimal accuracy varied from zero to one. In fact, large errors were generated for input coefficients with ranges of approximately 1.0E-2. It was found, in both cases, that the error generated increased with increasing matrix dimension. This is to be expected, since the number of calculations done in inverting the matrix is increased. For example, a 12 x 12 random matrix was inverted, and it was found that only four to five significant figures could be expected, vice the seven decimal accuracy in the 5 x 5 case.

One my circumvent the inaccuracies obtained when the input elements are range numbers, by the following device [4]. Suppose we wish to invert X, whose elements are range numbers. Take X_0 to be the matrix of mid-points of the elements of X, so that $X = X_0 + E$. The elements of E are range numbers of the form $[-e, e]$. Obtain X_0^{-1} using range arithmetic, and let

$$E' = [I - X_0^{-1}E + (X_0^{-1}E)^2 - \dots\dots\dots] X_0^{-1} (-EX_0^{-1}).$$

If the elements of E are small compared to X_0^{-1}, then the series will converge, and one can write

$$X^{-1} = X_0 + E'.$$

The results of a typical run for an inversion of a randomly generated 5 x 5 matrix are shown on the following page.

Element	Real Part	Lower	Upper
(1, 1)	4.6396189281E01	Same as Real	Same as Real
(1, 2)	8.6916184500E02		
(1, 3)	3.6039163981E02		
(1, 4)	5.8772778565E02		
(1, 5)	9.4693749440E02		
(2, 1)	5.5609401546E02		
(2, 2)	8.9089112203E02		
(2, 3)	4.0143081011E02		
(2, 4)	4.2098029158E02		
(2, 5)	6.1093177836E02		
(3, 1)	9.9636585643E02		
(3, 2)	1.6340010236E02		
(3, 3)	4.6658037215E02		
(3, 4)	5.0479887773E02		
(3, 5)	2.0507283101E02		
(4, 1)	2.7500284988E02		
(4, 2)	4.8510492363E02		
(4, 3)	2.4488146363E02		
(4, 4)	4.1884968178E02		
(4, 5)	1.9965179004E02		
(5, 1)	8.7792113349E02		
(5, 2)	2.3254672355E02		
(5, 3)	3.5002222285E00		
(5, 4)	3.2092384145E02		
(5, 5)	6.4308479314E02		

X - MATRIX

TABLE 2

We will present the real part, and the least significant digits of the lower and upper range numbers of X - INVERSE. The exponents will not be repeated.

Element	Real Part		Lower	Upper
(1, 1)	-9.1422040	728E-04	854	607
(1, 2)	7.419425	7112E-04	6975	7249
(1, 3)	9.3418389	435E-05	070	864
(1, 4)	-5.779019	5190E-05	5709	4688
(1, 5)	6.2948717	498E-04	477	517
(2, 1)	-1.122627	041E-03	052	032
(2, 2)	1.6441862	954E-03	944	965
(2, 3)	-1.3201931	162E-03	165	159
(2, 4)	1.4714936	870E-03	865	875
(2, 5)	5.5237176	832E-05	720	995
(3, 1)	1.0620289	855E-03	838	873
(3, 2)	1.1383145	460E-03	437	478
(3, 3)	2.3018511	807E-03	800	812
(3, 4)	-3.6993050	965E-03	973	956
(3, 5)	-2.2307817	966E-03	966	962
(4, 1)	6.4802494	170E-04	040	338
(4, 2)	-3.0018154	973E-03	990	955
(4, 3)	-5.102677	3200E-05	3669	2679
(4, 4)	4.0549999	820E-03	811	825
(4, 5)	6.5487719	463E-04	433	477
(5, 1)	1.3248530	755E-03	742	766
(5, 2)	-1.1561055	861E-04	968	706
(5, 3)	3.6280024	837E-04	794	865
(5, 4)	-2.4566800	524E-03	531	520
(5, 5)	3.6100559	502E-04	481	523

X - INVERSE

FIGURE 3

6. Conclusions.

Range arithmetic is a powerful tool, not only for error estimation in computations of a scientific or engineering nature, but for the evaluation and comparison of alternative numerical algorithms as well. The QRANGE7 package provides the user with a simple means for such analysis. The drawback in the use of QRANGE7, at this point, is that computing time is increased. There are many points where the program could be increased in efficiency, notably in the Q1Q04770 and Q1Q05770 subroutines for range multiplication and range division. The author feels, however, that even with the increase in computing time, the pay-off in the use of range arithmetic is so great that it more than outweighs the disadvantage.

A few function subprograms have been provided for use in conjunction with QRANGE7. They are used exactly as one would use the library subroutines in FORTRAN, except a "7" replaces the "F" at the end of the function name. One must remember to declare these functions TYPE RANGE7(3) before execution. Much work remains to be done in this area, and it is hoped that, eventually, a complete library of functions will be available for use with QRANGE7.

It might be noted, at this point, that there are errors generated in the conversion of decimal-to-binary and binary-to-decimal numbers during input/output, and these have not been taken into account. The author feels that these errors are negligible compared to the machine generated error occurring during computation.

The results obtained using QRANGE7 indicated that the technique of range arithmetic can, indeed, be used to provide error information at each stage of computation. Unfortunately, the author did not have the time to try it out on some of the more common numerical algorithms, such as the Runge - Kutta method for solving differential equations, or the Newton - Raphson method for determining roots of n^{th} order polynomials. It is hoped that this will be done in the future.

In conclusion, it is felt that range arithmetic is of such value that it warrants inclusion in future modifications of FORTRAN or other algebraic compilers as a standard TYPE.

BIBLIOGRAPHY

1. Moore, R. Lockheed Missiles - Document, LMSD-48421, 1959.

2. Shudde, R. A Computational Pitfall and Automatic Error Analysis, Atomics International Applied Mathematics Technical Document (AMTD-133), May, 1962.

3. Shudde, R. Input / Output for Multi-Word TYPE OTHER, Program Description 15/J5-NPGS-OTHERI/O (FTN-63), U. S. Naval Postgraduate School, April, 1966.

4. Collins, G. Interval Arithmetic for Automatic Error Analysis, Data Systems Division International Business Machine Corporation, Report M & A-5, 1960.

5. FORTRAN-63 / Reference Manual, Control Data Corporation, June, 1964.

APPENDIX I

QRANGE7 PROGRAM LISTING [5]

Entry Points:

Q1Q00770, Q1Q01770, Q1Q02770, Q1Q03770,

Q1Q04770, Q1Q05770, Q1Q10770, Q1Q00710,

Q1Q01710, Q1Q02710, Q1Q03710, Q1Q04710,

Q1Q05710, Q1Q10710, Q1Q10170, Q1Q00700,

Q1Q01700, Q1Q02700, Q1Q03700, Q1Q04700,

Q1Q05700, Q1Q10700, Q1Q10070, UNPACK7,

RNGAD7, RNGMU7, RNGDI7, QUIRKL7, QUIRKU7,

REPACK7, ERROR777

```
         IDENT     QRANGE7
RNG7S777 BLOCK     4
         COMMON    ACC(3),B
A        BSS       4
Q        BSS       4
T        BSS       3
A.       BSS       3
Q.       BSS       3
TEMP0    BSS       1
TEMP1    BSS       1
TEMP2    BSS       1
TEMP3    BSS       1
TEMP4    BSS       1
MTEMPU   BSS       1
DTEMPU   BSS       1
CA       BSS       1
CB       BSS       1
EA       BSS       1
EB       BSS       1
EL       BSS       1
EM1      BSS       1
EM2      BSS       1
FM1      BSS       1
FM2      BSS       1
BIT      BSS       1
S        BSS       1
C        BSS       1
AS       BSS       3
AS.      BSS       3
SA       BSS       3
AI       BSS       1
IA       BSS       1
AUX      BSS       3
AUX.     BSS       3
SAUX     BSS       3
```

```
SAUX.     BSS          3
QAUX      BSS          3
AF        BSS          1
          ENTRY        Q1Q00770                        THIS PROGRAM LOADS RANGE          1Q0000
Q1Q00770  SLJ          **                              ACCUMULATOR.   THE LOWER          1Q0010
          LDA          *                               GOES TO ACC+1,THE UPPER TO        1Q0020
          ALS          +24                             ACC+2, AND THE REAL PART          1Q0030
          INA          -1                              GOES TO ACC.                      1Q0040
          SAU          *+1                                                               1Q0050
+         ENA      7   **                                                                1Q0060
          SAU          A1                                                                1Q0070
          INA          +1                                                                1Q0080
          SAU          A2                                                                1Q0090
          INA          +1                                                                1Q0100
          SAU          A3                                                                1Q0110
A2        LDA          **                              LOAD LOWER RANGE NUMBER           1Q0120
          STA          ACC+1                           LOWER TO ACC+1                    1Q0130
A3        LDA          **                              LOAD UPPER RANGE NUMBER           1Q0140
          STA          ACC+2                           UPPER TO ACC+2                    1Q0150
A1        LDA          **                              LOAD REAL PART                    1Q0160
          STA          ACC                             REAL TO ACC                       1Q0170
          SLJ          Q1Q00770                                                          1Q0180
          ENTRY        Q1Q01770                        LOADS RANGE ACCUMULATOR           2Q0000
Q1Q01770  SLJ          **                              WITH THE COMPLEMENT OF A          2Q0010
          LDA          *                               RANGE NUMBER.                     2Q0020
          ALS          +24                             NOTE, WHEN LOWER AND UPPER        2Q0030
          INA          -1                              ARE COMPLEMENTED,THEIR ORDER      2Q0040
          SAU          *+1                             IS REVERSED.                      2Q0050
+         ENA      7   **                                                                2Q0060
          SAU          B1                                                                2Q0070
          INA          +1                                                                2Q0080
          SAU          B2                                                                2Q0090
          INA          +1                                                                2Q0100
          SAU          B3                                                                2Q0110
B2        LAC          **                              LOAD LOWER COMPLEMENT             2Q0120
```

```
          STA       ACC+2             STORE IN ACC+2 (NEW UPPER)         2Q0130
B3        LAC       **                LOAD UPPER COMPLEMENT              2Q0140
          STA       ACC+1             STORE IN ACC+1 (NEW LOWER)         2Q0150
B1        LAC       **                LOAD REAL COMPLEMENT               2Q0160
          STA       ACC               STORE IN ACC                       2Q0170
          SLJ       Q1Q01770                                             2Q0180
          ENTRY     Q1Q02770          ROUTINE FOR RANGE (+) RANGE,       3Q0000
Q1Q02770  SLJ       **                THAT IS  A7(+)B7. A7 IS IN         3Q0010
          LDA       *                 RANGE ACCUMULATOR.                 3Q0020
          ALS       +24                                                  3Q0030
          INA       -1                                                   3Q0040
          SAU       *+1                                                  3Q0050
+         ENA    7  **                                                   3Q0060
          SAL       C1                                                   3Q0070
          INA       +1                                                   3Q0080
          SAU       C2                                                   3Q0090
          INA       +1                                                   3Q0100
          SAU       C3                                                   3Q0110
          LDA       ACC+1             LOADS ACC+1 (A7 LOWER)             3Q0120
          STA       B                 STORES IN B                        3Q0130
C2        LDA       **                LOADS B7 LOWER                     3Q0140
          RTJ       UNPACK7           UNPACKS BOTH LOWER RANGE           3Q0150
+         RTJ       RNGAD7            NUMBERS,ADDS THEM                  3Q0160
+         RTJ       QUIRKL7           TRUNCATES UNPACKED RESULT, AND     3Q0170
+         RTJ       REPACK7           RENORMALIZES RESULT                3Q0180
+         STA       ACC+1             REPACKS AND STORES IN ACC+1        3Q0190
          LDA       ACC+2             LOADS ACC+2 (A7 UPPER)             3Q0200
          STA       B                 STORES IN B                        3Q0210
C3        LDA       **                LOADS B7 UPPER                     3Q0220
          RTJ       UNPACK7           UNPACKS A7 UPPER AND B7 UPPER      3Q0230
+         RTJ       RNGAD7            ADDS THEM                          3Q0240
+         RTJ       QUIRKU7           ROUNDS AND RENORMALIZES RESULT     3Q0250
+         RTJ       REPACK7           REPACKS RESULT AND                 3Q0260
+         STA       ACC+2             STORES IN ACC+2                    3Q0270
C1        LDA       ACC               LOADS A7 REAL                      3Q0280
```

```
         FAD        **          FLOATING ADDS B7 REAL              3Q0290
         STA        ACC         STORES RESULT IN ACC               3Q0300
         SLJ        Q1Q02770                                       3Q0310
         ENTRY      Q1Q03770    ROUTINE FOR RANGE (-) RANGE,       4Q0000
Q1Q03770 SLJ        **          I.E. A7(-)B7. A7 IN RANGE          4Q0010
         LDA        *           ACCUMULATOR.                       4Q0020
         ALS        +24                                            4Q0030
         INA        -1                                             4Q0040
         SAU        *+1                                            4Q0050
+        ENA     7  **                                             4Q0060
         SAL        D1                                             4Q0070
         INA        +1                                             4Q0080
         SAU        D2                                             4Q0090
         INA        +1                                             4Q0100
         SAU        D3                                             4Q0110
         LDA        ACC+2       LOAD A7 UPPER                      4Q0120
         STA        B           STORE IN B                         4Q0130
D2       LAC        **          LOAD COMPLEMENT OF B7 LOWER        4Q0140
+        RTJ        UNPACK7     UNPACK, ADD, ROUND,RENORMALIZE     4Q0150
+        RTJ        RNGAD7      RESULT                             4Q0160
+        RTJ        QUIRKU7                                        4Q0170
+        RTJ        REPACK7                                        4Q0180
+        STA        ACC+2       RESULT TO ACC+2                    4Q0190
         LDA        ACC+1       LOAD A7 LOWER                      4Q0200
         STA        B           STORE IN B                         4Q0210
D3       LAC        **          LOAD B7 UPPER COMPLEMENT           4Q0220
+        RTJ        UNPACK7     PERFORM RANGE ADDITION             4Q0230
+        RTJ        RNGAD7                                         4Q0240
+        RTJ        QUIRKL7                                        4Q0250
+        RTJ        REPACK7                                        4Q0260
+        STA        ACC+1       STORE IN ACC+1                     4Q0270
D1       LDA        ACC         LOAD A7 REAL                       4Q0280
         FSB        **          FLOATING SUBTRACT B7 REAL          4Q0290
         STA        ACC         RESULT TO ACC                      4Q0300
         SLJ        Q1Q03770                                       4Q0310
```

```
         ENTRY      Q1Q04770                     ROUTINE FOR RANGE (*) RANGE,      5Q0000
Q1Q04770 SLJ        **                           I.E. A7*B7. A7 IN RANGE ACC.      5Q0010
         LDA        *                                                              5Q002
         ALS        +24                                                            5Q0030
         INA        -1                                                             5Q0040
         SAU        *+1                                                            5Q0050
+        ENA     7  **                                                             5Q0060
         SAL        RE                                                             5Q0070
         INA        +1                                                             5Q0080
         SAU        E21                                                            5Q0090
         SAU        E22                                                            5Q0100
         INA        +1                                                             5Q0110
         SAU        E31                                                            5Q0120
         SAU        E32                                                            5Q0130
E21      LDA        **                           CALCULATES ALL POSSIBLE PRODUCTS  5Q0140
         STA        EM1                          USING REAL ARITHMETIC             5Q0150
         FMU        ACC+1                                                          5Q0160
         STA        A                            A7 LOWER(*)B7 LOWER. STORE IN A.  5Q0170
E22      LDA        **                           A7 UPPER(*)B7 LOWER.              5Q0180
         FMU        ACC+2                        STORE IN A+1                      5Q0190
         STA        A+1                                                            5Q0200
E31      LDA        **                                                             5Q0210
         STA        EM2                                                            5Q0220
         FMU        ACC+1                        A7 LOWER(*)B7 UPPER               5Q0230
         STA        A+2                          STORE IN A+2                      5Q0240
E32      LDA        **                                                             5Q0250
         FMU        ACC+2                        A7 UPPER(*)B7 UPPER               5Q0260
         STA        A+3                          STORE IN A+3                      5Q0270
         LDA        A                            THIS PORTION OF ROUTINE SEARCHES  5Q0280
         FSB        A+1                          FOR THE MAXIMUM OF THE FOUR       5Q0290
         AJP     2  ETST1                        PRODUCTS                          5Q0300
         LDA        A+1                                                            5Q0310
         FSB        A+2                                                            5Q0320
         AJP     2  ETST3                                                          5Q0330
         LDA        A+2                                                            5Q0340
```

```
         FSB      A+3                                        5Q0350
         AJP   2  RMAX3                                      5Q0360
         SLJ      RMAX4                                      5Q0370
ETST1    LDA      A                                          5Q0380
         FSB      A+2                                        5Q0390
         AJP   2  ETST2                                      5Q0400
         LDA      A+2                                        5Q0410
         FSB      A+3                                        5Q0420
         AJP   2  RMAX3                                      5Q0430
         SLJ      RMAX4                                      5Q0440
ETST2    LDA      A                                          5Q0450
         FSB      A+3                                        5Q0460
         AJP   2  RMAX1                                      5Q0470
         SLJ      RMAX4                                      5Q0480
ETST3    LDA      A+1                                        5Q0490
         FSB      A+3                                        5Q0500
         AJP   2  RMAX2                                      5Q0510
         SLJ      RMAX4                                      5Q0520
RMAX1    LDA      ACC+1         MAX IN A                     5Q0530
         STA      B                                          5Q054
         LDA      EM1                                        5Q0550
         SLJ      M                                          5Q056
RMAX2    LDA      ACC+2         MAX IN A+1                   5Q0570
         STA      B                                          5Q058
         LDA      EM1                                        5Q0590
         SLJ      M                                          5Q060
RMAX3    LDA      ACC+1         MAX IN A+2                   5Q0610
         STA      B                                          5Q062
         LDA      EM2                                        5Q0630
         SLJ      M                                          5Q064
RMAX4    LDA      ACC+2         MAX IN A+3                   5Q0650
         STA      B                                          5Q066
         LDA      EM2                                        5Q0670
         SLJ      M                                          5Q068
M        RTJ      UNPACK7       RECALCULATE MAX PRODUCT USING 5Q0690
```

```
+        RTJ       RNGMU7            RANGE ARITHMETIC              5Q0700
+        RTJ       QUIRKU7                                         5Q0710
+        RTJ       REPACK7                                         5Q0720
+        STA       MTEMPU            STORE IN MTEMPU               5Q0740
         LDA       A                 FIND MIN PRODUCT              5Q0750
         FSB       A+1                                             5Q0760
         AJP    3  EMST1                                           5Q0770
         LDA       A+1                                             5Q0780
         FSB       A+2                                             5Q0790
         AJP    3  EMST3                                           5Q0800
         LDA       A+2                                             5Q0810
         FSB       A+3                                             5Q0820
         AJP    3  RMIN3                                           5Q0830
         SLJ       RMIN4                                           5Q0840
EMST1    LDA       A                                               5Q0850
         FSB       A+2                                             5Q0860
         AJP    3  EMST2                                           5Q0870
         LDA       A+2                                             5Q0880
         FSB       A+3                                             5Q0890
         AJP    3  RMIN3                                           5Q0900
         SLJ       RMIN4                                           5Q0910
EMST2    LDA       A                                               5Q0920
         FSB       A+3                                             5Q0930
         AJP    3  RMIN1                                           5Q0940
         SLJ       RMIN4                                           5Q0950
EMST3    LDA       A+1                                             5Q0960
         FSB       A+3                                             5Q0970
         AJP    3  RMIN2                                           5Q0980
         SLJ       RMIN4                                           5Q0990
RMIN1    LDA       ACC+1             MIN IN A.                     5Q1000
         STA       B                                               5Q101
         LDA       EM1                                             5Q1020
         SLJ       NM                                              5Q1030
RMIN2    LDA       ACC+2             MIN IN A+1                    5Q1040
```

```
         STA     B                                                5Q105
         LDA     EM1                                              5Q1060
         SLJ     NM                                               5Q1070
RMIN3    LDA     ACC+1              MIN IN A+2                    5Q1080
         STA     B                                                5Q109
         LDA     EM2                                              5Q1100
         SLJ     NM                                               5Q1110
RMIN4    LDA     ACC+2              MIN IN A+3                    5Q1120
         STA     B                                                5Q113
         LDA     EM2                                              5Q1140
NM       RTJ     UNPACK7            RECALCULATE MIN PRODUCT USING 5Q1150
+        RTJ     RNGMU7             RANGE ARITHMETIC              5Q1160
+        RTJ     QUIRKL7                                          5Q1170
+        RTJ     REPACK7                                          5Q1180
+        STA     ACC+1              STORE IN ACC+1                5Q1200
         LDA     MTEMPU             LOAD MTEMPU                   5Q1201
         STA     ACC+2              STORE IN ACC+2                5Q1202
RE       LDA     ACC                                              5Q1210
         FMU     **                 CALCULATE REAL PRODUCT        5Q1220
         STA     ACC                STORE IN ACC                  5Q1230
         SLJ     Q1Q04770                                         5Q1240
         ENTRY   Q1Q05770           ROUTINE FOR RANGE(/)RANGE,    6Q0000
Q1Q05770 SLJ     **                 I.E. A7/B7. A7 IN RANGE ACC   6Q0010
         LDA     *                                                6Q002
         ALS     +24                                              6Q0030
         INA     -1                                               6Q0040
         SAU     *+1                                              6Q0050
+        ENA   7 **                                               6Q0060
         SAL     FPACKUP                                          6Q0070
         SAL     FERROR+1                                         6Q0080
         INA     +1                                               6Q0090
         SAU     FTESTLO                                          6Q0100
         SAU     FOKAY                                            6Q0110
         SAL     F21                                              6Q0120
         SAL     F22                                              6Q0130
```

```
         INA       +1                                                   6Q0140
         SAU       FTESTHI                                              6Q0150
         SAU       FSTORE                                               6Q0160
         SAL       F31                                                  6Q0170
         SAL       F32                                                  6Q0180
FTESTLO  LDA       **             TESTS TO SEE IF DIVISOR               6Q0190
         AJP     2 FOKAY          BRACKETS ZERO, IF SO ERROR777         6Q0200
FTESTHI  LDA       **             IS CALLED AND DIAGNOSTIC IS           6Q0210
         AJP     3 FOKAY          PRINTED. IF NOT, ROUTINE IS           6Q0220
FERROR   RTJ       ERROR777       EXECUTED                              6Q0230
+        0         Q1005770                                             6Q0240
         0         **                                                   6Q0250
         BCD       1DIVISION                                            6Q0260
FOKAY    LDA       **             CALCULATE ALL POSSIBLE QUOTIENTS      6Q0270
         STA       FM1                                                  6Q0280
F21      LDA       ACC+1                                                6Q0290
         FDV       **                                                   6Q0300
         STA       A              A7 LOWER(/)B7 LOWER IN A              6Q0310
F22      LDA       ACC+2                                                6Q0320
         FDV       **                                                   6Q0330
         STA       A+1            A7 UPPER(/)B7 LOWER IN A+1            6Q0340
FSTORE   LDA       **                                                   6Q0350
         STA       FM2                                                  6Q0360
F31      LDA       ACC+1                                                6Q0370
         FDV       **                                                   6Q0380
         STA       A+2            A7 LOWER(/)B7 UPPER IN A+2            6Q0390
F32      LDA       ACC+2                                                6Q0400
         FDV       **                                                   6Q0410
         STA       A+3            A7 UPPER(/)B7 UPPER IN A+3            6Q0420
         LDA       A              SEARCH FOR MAXIMUM QUOTIENT           6Q0430
         FSB       A+1                                                  6Q0440
         AJP     2 FTST1                                                6Q0450
         LDA       A+1                                                  6Q0460
         FSB       A+2                                                  6Q0470
         AJP     2 FTST3                                                6Q0480
```

```
         LDA       A+2                                      6Q0490
         FSB       A+3                                      6Q0500
         AJP    2  FMAX3                                    6Q0510
         SLJ       FMAX4                                    6Q0520
FTST1    LDA       A                                        6Q0530
         FSB       A+2                                      6Q0540
         AJP    2  FTST2                                    6Q0550
         LDA       A+2                                      6Q0560
         FSB       A+3                                      6Q0570
         AJP    2  FMAX3                                    6Q0580
         SLJ       FMAX4                                    6Q0590
FTST2    LDA       A                                        6Q0600
         FSB       A+3                                      6Q0610
         AJP    2  FMAX1                                    6Q0620
         SLJ       FMAX4                                    6Q0630
FTST3    LDA       A+1                                      6Q0640
         FSB       A+3                                      6Q0650
         AJP    2  FMAX2                                    6Q0660
         SLJ       FMAX4                                    6Q0670
FMAX1    LDA       FM1          MAX IN A                    6Q0680
         STA       B                                        6Q069
         LDA       ACC+1                                    6Q0700
         SLJ       MAXR                                     6Q0710
FMAX2    LDA       FM1          MAX IN A+1                  6Q0720
         STA       B                                        6Q073
         LDA       ACC+2                                    6Q0740
         SLJ       MAXR                                     6Q0750
FMAX3    LDA       FM2          MAX IN A+2                  6Q0760
         STA       B                                        6Q077
         LDA       ACC+1                                    6Q0780
         SLJ       MAXR                                     6Q0790
FMAX4    LDA       FM2          MAX IN A+3                  6Q0800
         STA       B                                        6Q081
         LDA       ACC+2                                    6Q0820
MAXR     RTJ       UNPACK7      RECOMPUTE MAX QUOTIENT USING 6Q0840
```

```
+         RTJ        RNGDI7              UN-NORMALIZED RANGE ARITHMETIC      6Q0850
+         RTJ        QUIRKU7                                                 6Q0860
+         RTJ        REPACK7                                                 6Q0870
+         STA        DTEMPU              STORE QUOTIENT UPPER IN DTEMPU      6Q0890
          LDA        A                   SEARCH FOR MIN QUOTIENT             6Q0900
          FSB        A+1                                                     6Q0910
          AJP     3  MFTST1                                                  6Q0920
          LDA        A+1                                                     6Q0930
          FSB        A+2                                                     6Q0940
          AJP     3  MFTST3                                                  6Q0950
          LDA        A+2                                                     6Q0960
          FSB        A+3                                                     6Q0970
          AJP     3  FMIN3                                                   6Q0980
          SLJ        FMIN4                                                   6Q0990
MFTST1    LDA        A                                                       6Q1000
          FSB        A+2                                                     6Q1010
          AJP     3  MFTST2                                                  6Q1020
          LDA        A+2                                                     6Q1030
          FSB        A+3                                                     6Q1040
          AJP     3  FMIN3                                                   6Q1050
          SLJ        FMIN4                                                   6Q1060
MFTST2    LDA        A                                                       6Q1070
          FSB        A+3                                                     6Q1080
          AJP     3  FMIN1                                                   6Q1090
          SLJ        FMIN4                                                   6Q1100
MFTST3    LDA        A+1                                                     6Q1110
          FSB        A+3                                                     6Q1120
          AJP     3  FMIN2                                                   6Q1130
          SLJ        FMIN4                                                   6Q1140
FMIN1     LDA        FM1                 MIN IN A                            6Q1150
          STA        B                                                       6Q116
          LDA        ACC+1                                                   6Q1170
          SLJ        MINR                                                    6Q1180
FMIN2     LDA        FM1                 MIN IN A+1                          6Q1190
```

```
         STA      B                                            6Q120
         LDA      ACC+2                                        6Q1210
         SLJ      MINR                                         6Q1220
FMIN3    LDA      FM2              MIN IN A+2                  6Q1230
         STA      B                                            6Q124
         LDA      ACC+1                                        6Q1250
         SLJ      MINR                                         6Q1260
FMIN4    LDA      FM2              MIN IN A+3                  6Q1270
         STA      B                                            6Q128
         LDA      ACC+2                                        6Q1290
MINR     RTJ      UNPACK7          RECOMPUTE MIN QUOTIENT USING  6Q1310
+        RTJ      RNGDI7           UN-NORMALIZED RANGE ARITHMETIC 6Q1320
+        RTJ      QUIRKL7                                      6Q1330
+        RTJ      REPACK7                                      6Q1340
+        STA      ACC+1            STORE QUOTIENT LOWER IN ACC+1 6Q1360
         LDA      DTEMPU                                       6Q1361
         STA      ACC+2            STORE QUOTIENT UPPER IN ACC+2 6Q1362
FPACKUP  LDA      ACC              CALCULATE REAL QUOTIENT     6Q1370
         FDV      **                                           6Q1380
         STA      ACC              STORE IN ACC                6Q1390
         SLJ      Q1Q05770                                     6Q1400
         ENTRY    ERROR777                                     1E0000
...ADD.. OCT      0                                            1E0010
FORMAT.  BSS      15                                           1E0020
ERROR777 SLJ      **                                           1E0030
         LDA      *                                            1E004
         ALS      +24                                          1E0050
         SAU      H                                            1E006
         INA      +1                                           1E0070
         SAL      H2                                           1E0080
         INA      +1                                           1E0090
         SAU      H3                                           1E0100
H0       LDA      **                                           1E0110
         SAL      H2+1                                         1E0120
         ALS      +24                                          1E0130
```

```
         SAU       *+1                                              1E0140
+        LDA       **                                               1E0150
         ALS       +24                                              1E0160
         INA       -1                                               1E0170
H1       SAL       ...ADD..                                         1E0180
         ENA       +51                                              1E0190
         ENQ       ..2000                                           1E0200
+        CALL      Q8QINGOT                                         1E0210
         0         0                                                1E022
         CALL      Q8QGOTTY                                         1E0230
         0         GG                                               1E0240
H2       0         0                                                1E0250
         1         **                                               1E0260
+        60        +3                                               1E0270
         1         **                                               1E0280
         0         0                                                1E029
         1         ...ADD..                                         1E0300
GG       CALL      Q8QENGOT                                         1E0310
         ORGR      FORMAT.                                          1E0320
..2000   BCD       5(20HOERROR IN RANGE7(3) ,A8,28H ROUTINE.        1E0330
         BCD       5 EXECUTION DELETED.//12H ARGUMENT = 3E25        1E0340
         BCD       5 .10//22H CALLED FROM LOCATION O5)              1E0350
         ORGR      *                                                1E036
+        CALL      ERROR*                                           1E0370
H3       SLJ       **                                               1E0380
         ENTRY     Q1Q10770          ROUTINE TO STORE RANGE ACC.    7Q0000
Q1Q10770 SLJ       **                                               7Q0010
         LDA       *                                                7Q002
         ALS       +24                                              7Q0030
         INA       -1                                               7Q0040
         SAU       *+1                                              7Q0050
+        ENA     7 **                                               7Q0060
         SAL       G1                                               7Q0070
         INA       +1                                               7Q0080
         SAL       G2                                               7Q0090
```

```
          INA      +1                                           7Q0100
          SAL      G3                                           7Q0110
G2        LDA      ACC+1            STORE RANGE LOWER (ACC+1)   7Q0120
          STA      **                                           7Q0130
G3        LDA      ACC+2            STORE RANGE UPPER (ACC+2)   7Q0140
          STA      **                                           7Q0150
G1        LDA      ACC              STORE REAL PART  (ACC)      7Q0160
          STA      **                                           7Q0170
          SLJ      Q1Q10770                                     7Q0180
          ENTRY    Q1Q00710         ROUTINE TO LOAD REAL INTO   8Q0000
Q1Q00710  SLJ      **               RANGE ACCUMULATOR           8Q0010
          LDA      *                                            8Q002
          ALS      +24                                          8Q0030
          INA      -1                                           8Q0040
          SAU      *+1                                          8Q0050
+         ENA    7 **                                           8Q0060
          SAU      AA1                                          8Q0070
          SAU      AA2                                          8Q0080
          SAU      AA3                                          8Q0090
AA2       LDA      **               LOAD REAL                   8Q0100
          STA      ACC+1            STORE IN ACC+1              8Q0110
          ENA      0                                            8Q012
          STA      B                                            8Q013
AA3       LDA      **               LOAD REAL                   8Q0140
          RTJ      UNPACK7          UN-NORMALIZE ADD ZERO,USING 8Q0150
+         RTJ      RNGAD7           QUIRKU7 ROUTINE FOR RANGE UPPER 8Q0160
+         RTJ      QUIRKU7                                      8Q0170
+         RTJ      REPACK7                                      8Q0180
+         STA      ACC+2            STORE IN ACC+2              8Q0190
AA1       LDA      **               LOAD REAL                   8Q0200
          STA      ACC              STORE IN ACC.               8Q0210
          SLJ      Q1Q00710                                     8Q0220
          ENTRY    Q1Q01710                                     9Q0000
Q1Q01710  SLJ      **               ROUTINE TO LOAD REAL COMPLEMENT 9Q0010
          LDA      *                INTO RANGE ACCUMULATOR      9Q0020
```

```
         ALS       +24                                              9Q0030
         INA       -1                                               9Q0035
         SAU       *+1                                              9Q0040
+        ENA     7 **                                               9Q0050
         SAU       *+1                                              9Q0060
         LAC       **            LOAD REAL COMPLEMENT               9Q0070
         STA       C             STORE TEMPORARILY IN C             9Q0080
         RTJ       Q1Q00710      CALL ROUTINE TO LOAD C INTO        9Q0090
         0         C             RANGE ACCUMULATOR                  9Q0100
         SLJ       Q1Q01710                                         9Q0110
         ENTRY     Q1Q02710                                        10Q0000
Q1Q02710 SLJ       **            ROUTINE FOR RANGE(+)REAL          10Q0010
         LDA       *             RANGE NUMBER IS IN RANGE          10Q0020
         ALS       +24           ACCUMULATOR                       10Q0030
         INA       -1                                              10Q0040
         SAU       *+1                                             10Q0050
+        ENA     7 **                                              10Q0060
         SAL       *+2                                             10Q0070
+        RTJ       Q1Q10770      STORE RANGE ACCUMULATOR IN AS     10Q0080
         0         AS                                              10Q0090
+        RTJ       Q1Q00710      CONVERT REAL TO RANGE AND         10Q0100
         0         **            PUT IN RANGE ACCUMULATOR          10Q0110
         RTJ       Q1Q02770      CALL ROUTINE FOR RANGE(+)RANGE    10Q0120
         0         AS            ADD.  RESULT IS IN RANGE          10Q0130
         SLJ       Q1Q02710      ACCUMULATOR                       10Q0140
         ENTRY     Q1Q03710                                        11Q0000
Q1Q03710 SLJ       **            ROUTINE FOR RANGE(-)REAL          11Q0010
         LDA       *             RANGE NUMBER IS IN RANGE          11Q0020
         ALS       +24           ACCUMULATOR                       11Q0030
         INA       -1                                              11Q0040
         SAU       *+1                                             11Q0050
+        ENA     7 **                                              11Q0060
         SAL       *+2                                             11Q0070
+        RTJ       Q1Q10770      STORE RANGE NUMBER IN   A.        11Q0080
         0         A.                                              11Q0090
```

```
+         RTJ       Q1Q01710          LOAD REAL COMPLEMENT INTO         11Q0100
          O         **                RANGE ACCUMULATOR                 11Q0110
          RTJ       Q1Q02770          ADD THE TWO RANGE NUMBERS         11Q0120
          O         A.                RESULT IS IN RANGE ACCUMULATOR    11Q0130
          SLJ       Q1Q03710                                            11Q0140
          ENTRY     Q1Q04710                                            12Q0000
Q1Q04710  SLJ       **                ROUTINE FOR RANGE(*)REAL          12Q0010
          LDA       *                                                   12Q002
          ALS       +24                                                 12Q0030
          INA       -1                                                  12Q0040
          SAU       *+1                                                 12Q0050
+         ENA     7 **                                                  12Q0060
          SAL       *+2                                                 12Q0070
+         RTJ       Q1Q10770          STORE RANGE ACCUMULATOR INTO      12Q0080
          O         AS.               AS.                               12Q0090
+         RTJ       Q1Q00710          LOAD REAL INTO RANGE ACCUMULATOR  12Q0100
          O         **                                                  12Q0110
          RTJ       Q1Q04770          MULTIPLY RANGE NUMBERS            12Q0120
          O         AS.               REAULT IN RANGE ACCUMULATOR       12Q0130
          SLJ       Q1Q04710                                            12Q0140
          ENTRY     Q1Q05710                                            13Q0000
Q1Q05710  SLJ       **                ROUTINE FOR RANGE(/)REAL          13Q0010
          LDA       *                                                   13Q002
          ALS       +24                                                 13Q0030
          INA       -1                                                  13Q0040
          SAU       *+1                                                 13Q0050
+         ENA     7 **                                                  13Q0060
          SAL       *+2                                                 13Q0070
+         RTJ       Q1Q10770          STORE RANGE ACCUMULATOR INTO      13Q0080
          O         Q.                Q.                                13Q0090
+         RTJ       Q1Q00710          LOAD REAL INTO RANGE              13Q0100
          O         **                ACCUMULATOR                       13Q0110
          RTJ       Q1Q10770          STORE RANGE ACCUMULATOR INTO      13Q0120
          O         SA                SA                                13Q0130
          RTJ       Q1Q00770          LOAD RANGE ACCUMULATOR FROM       13Q0140
          O         Q.                Q.                                13Q0150
```

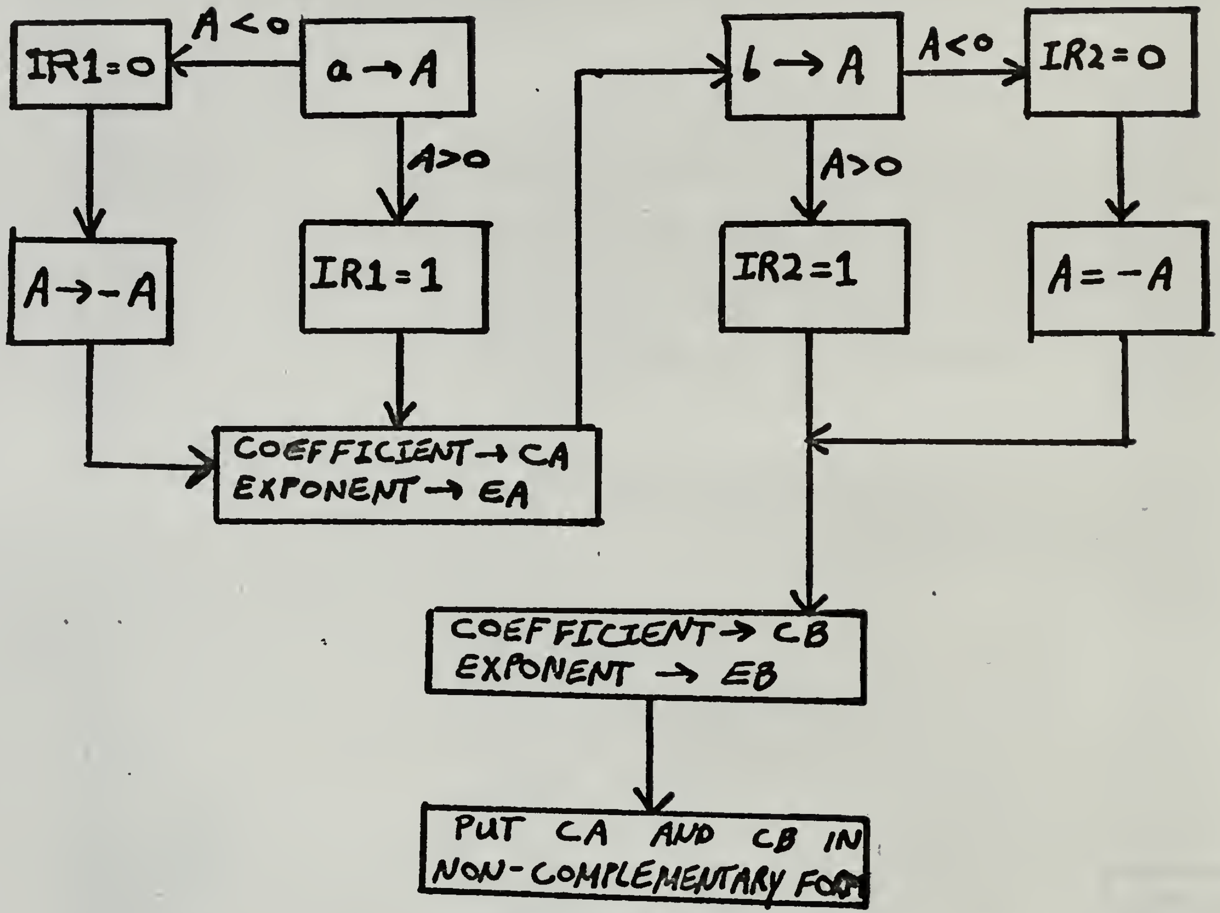
```
        RTJ        Q1Q05770         RANGE DIVIDE BY RANGE NUMBER       13Q00160
        0          SA               IN SA                              13Q00170
        SLJ        Q1Q05710         RESULT IN RANGE ACCUMULATOR        13Q00180
        ENTRY      Q1Q10710                                            14Q00000
Q1Q10710 SLJ       **               STORE RANGE INTO REAL              14Q00010
        LDA        *                                                   14Q002
        ALS        +24                                                 14Q00030
        INA        -1                                                  14Q00040
        SAU        *+1                                                 14Q00050
+       ENA     7  **                                                  14Q00060
        SAL        *+1                                                 14Q00070
+       LDA        ACC              LOAD ACC(REAL PART OF RANGE NO.)   14Q00080
        STA        **               STORE ACC                          14Q00090
        SLJ        Q1Q10710                                            14Q00100
        ENTRY      Q1Q10170                                            15Q00000
Q1Q10170 SLJ       **               STORE REAL INTO RANGE              15Q00010
        STA        ACC              STORE REAL INTO ACC                15Q00020
        STA        ACC+1            STORE REAL INTO ACC+1              1580030
        ENA        +                PUT ZERO IN A-REGISTER             1580040
        STA        B                STORE IN B                         1580050
+       LDA        ACC              LOAD ACC                           1580060
        RTJ        UNPACK7          UN-NORMALIZED ADD ZERO TO          1580070
+       RTJ        RNGAD7           ACC, CONSIDERING IT TO BE          1580080
+       RTJ        QUIRKU7          ADDITION OF UPPER RANGE            15Q00090
+       RTJ        REPACK7          NUMBERS                            15Q00100
+       STA        ACC+2            STORE IN ACC+2                     1580110
+       LDA        Q1Q10170                                            15Q00120
        ALS        +24                                                 1580130
        INA        -1                                                  15Q00140
        SAU        *+1                                                 15Q00150
+       ENA     7  **                                                  15Q00160
        SAL        *+1                                                 15Q00170
        RTJ        Q1Q10770         STORE RANGE ACCUMULATOR            15Q00180
        0          **                                                  15Q00190
        SLJ        Q1Q10170                                            15Q00200
        ENTRY      Q1Q00700                                            16Q00000
```

```
Q1Q00700 SLJ        **           LOAD INTEGER INTO RANGE              16Q0010
         LDA        *                                                 16Q002
         ALS        +24                                               16Q0030
         INA        -1                                                16Q0040
         SAU        *+1                                               16Q0050
+        LDA     7  **           LOAD INTEGER                         16Q0060
         CALL       FLOATF       CONVERT TO FLOATING POINT(REAL)      16Q0070
+        STA        AI           STORE IN AI                          16Q0080
+        RTJ        Q1Q00710     CALL ROUTINE TO LOAD  A(REAL)        16Q0090
         0          AI           INTO RANGE                           16Q0100
         SLJ        Q1Q00700                                          16Q0110
         ENTRY      Q1Q01700                                          17Q0000
Q1Q01700 SLJ        **           LOAD INTEGER COMPLEMENT INTO         17Q0010
         LDA        *            RANGE                                17Q0020
         ALS        +24                                               17Q0030
         INA        -1                                                17Q0040
         SAU        *+1                                               17Q0050
+        LAC     7  **           LOAD INTEGER COMPLEMENT              17Q0060
         CALL       FLOATF       CONVERT TO REAL                      17Q0070
+        STA        IA           STORE IN IA                          17Q0080
+        RTJ        Q1Q00710     LOAD REAL TO RANGE                   17Q0090
         0          IA                                                17Q0100
         SLJ        Q1Q01700                                          17Q0110
         ENTRY      Q1Q02700                                          18Q0000
Q1Q02700 SLJ        **           ROUTINE FOR RANGE(+)INTEGER          18Q0010
         LDA        *                                                 18Q002
         ALS        +24                                               18Q0030
         INA        -1                                                18Q0040
         SAU        *+1                                               18Q0050
+        ENA     7  **                                                18Q0060
         SAL        *+2                                               18Q0070
+        RTJ        Q1Q10770     STORE RANGE ACCUMULATOR              18Q0080
         0          AUX          IN AUX                               18Q0090
+        RTJ        Q1Q00700     LOAD INTEGER TO RANGE                18Q0100
         0          **           ADD THE RANGE NUMBERS                18Q0110
         RTJ        Q1Q02770                                          18Q0120
         0          AUX                                               18Q0130
```

```
         SLJ        Q1Q02700                                      18Q0140
         ENTRY      Q1Q03700                                      19Q0000
Q1Q03700 SLJ        **               ROUTINE FOR RANGE(-)INTEGER  19Q0010
         LDA        *                                             19Q002
         ALS        +24                                           19Q0030
         INA        -1                                            19Q0040
         SAU        *+1                                           19Q0050
+        ENA     7  **                                            19Q0060
         SAL        *+2                                           19Q0070
+        RTJ        Q1Q10770         STORE RANGE ACCUMULATOR      19Q0080
         0          AUX.             IN AUX.                      19Q0090
+        RTJ        Q1Q01700         LOAD INTEGER COMPLEMENT TO   19Q0100
         0          **               RANGE                        19Q0110
         RTJ        Q1Q02770         ADD RANGE NUMBERS            19Q0120
         0          AUX.                                          19Q0130
         SLJ        Q1Q03700                                      19Q0140
         ENTRY      Q1Q04700                                      20Q0000
Q1Q04700 SLJ        **               ROUTINE FOR RANGE(*)INTEGER  20Q0010
         LDA        *                                             20Q002
         ALS        +24                                           20Q0030
         INA        -1                                            20Q0040
         SAU        *+1                                           20Q0050
+        ENA     7  **                                            20Q0060
         SAL        *+2                                           20Q0070
+        RTJ        Q1Q10770         STORE RANGE ACCUMULATOR      20Q0080
         0          SAUX             IN SAUX                      20Q0090
+        RTJ        Q1Q00700         LOAD INTEGER TO RANGE        20Q0100
         0          **                                            20Q0110
         RTJ        Q1Q04770         MULTIPLY RANGE NUMBERS       20Q0120
         0          SAUX                                          20Q0130
         SLJ        Q1Q04700                                      20Q0140
         ENTRY      Q1Q05700                                      21Q0000
Q1Q05700 SLJ        **               ROUTINE FOR RANGE(/)INTEGER  21Q0010
         LDA        *                                             21Q002
         ALS        +24                                           21Q0030
```

```
          INA       -1                                                    21Q00040
          SAU       *+1                                                   21Q00050
+         ENA    7  **                                                    21Q00060
          SAL       *+2                                                   21Q00070
+         RTJ       Q1Q10770             STORE RANGE ACCUMULATOR          21Q00080
          0         QAUX                 IN QAUX                          21Q00090
+         RTJ       Q1Q00700             LOAD INTEGER TO RANGE            21Q00100
          0         **                                                    21Q00110
          RTJ       Q1Q10770             STORE RANGE ACCUMULATOR          21Q00120
          0         SAUX.                IN SAUX.                         21Q00130
          RTJ       Q1Q00770             LOAD RANGE ACCUMULATOR           21Q00140
          0         QAUX                 FROM QAUX                        21Q00150
          RTJ       Q1Q05770             DIVIDE RANGE BY RANGE NUMBER     21Q00160
          0         SAUX.                IN SAUX.                         21Q00170
          SLJ       Q1Q05700                                              21Q00180
          ENTRY     Q1Q10700                                              22Q00000
Q1Q10700  SLJ       **                   STORE RANGE INTO INTEGER         22Q00010
          LDA       *                                                     22Q002
          ALS       +24                                                   22Q00030
          INA       -1                                                    22Q00040
          SAU       *+1                                                   22Q00050
+         ENA    7  **                                                    22Q00055
          SAU       *+2                                                   22Q00060
          LDA       ACC                  LOAD ACC (REAL PART)             22Q00070
          CALL      XINTF                CONVERT TO INTEGER               22Q00080
+         STA       **                   STORE INTEGER                    22Q00090
          SLJ       Q1Q10700                                              22Q00100
          ENTRY     Q1Q10070                                              23Q00000
Q1Q10070  SLJ       **                   STORE INTEGER INTO RANGE         23Q00010
          CALL      FLOATF               CONVERT INTEGER TO REAL          23Q00020
+         STA       AF                   STORE IN AF                      23Q00030
          LDA       *-1                                                   23Q00040
          ALS       +24                                                   23Q00050
          INA       -1                                                    23Q00060
          SAU       *+1                                                   23Q00070
```

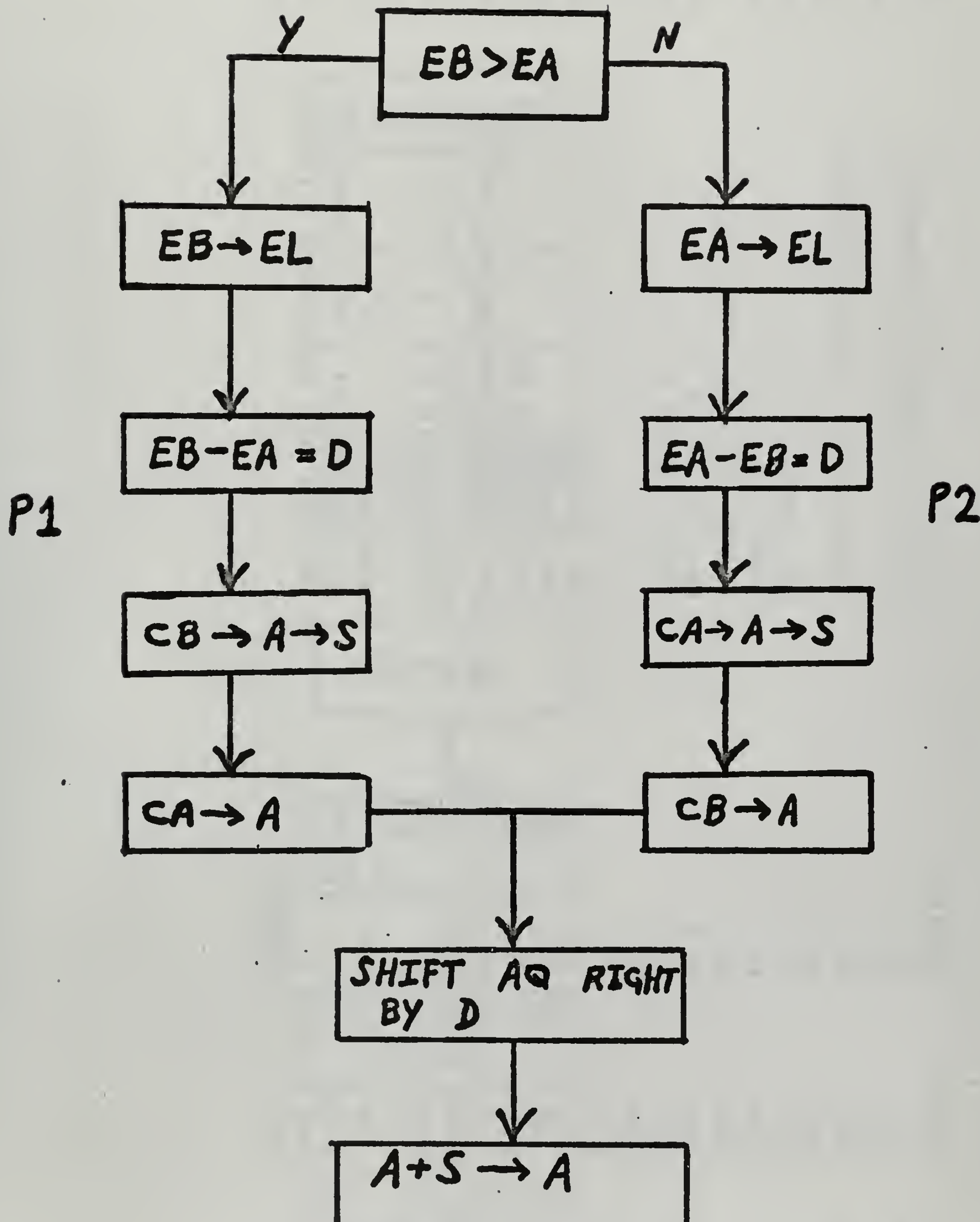
```
+      ENA    7  **                                     23Q00080
       SAL       *+2                                    23Q00090
+      LDA       AF             LOAD AF                 23Q00100
+      RTJ       Q1Q10170       STORE REAL  INTO RANGE  23Q00110
       0         **                                     23Q00120
       SLJ       Q1Q10070                               23Q00130
```



FLOW CHART FOR UNPACK7

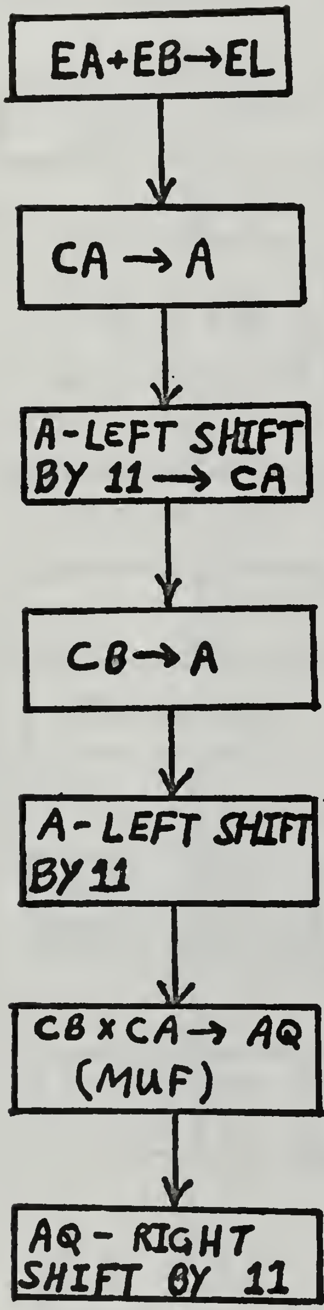
```
         ENTRY      UNPACK7                                               1UN0000
UNPACK7  SLJ        **                    UNPACKS TWO FLOATING POINT      1UN0001
         SIU     1  T22                   NUMBERS, A AND B                1UN0002
         SIL     2  T22                                                   1UN0010
         STA        TEMPO                 UNPACK A                        1UN0020
         LDQ        TEMPO                                                 1UN0030
         QJP     M  T3                                                    1UN0040
         ENI     1  1                                                     1UN005
         SLJ        T4                                                    1UN0060
T3       LQC        TEMPO                                                 1UN0070
         ENI     1  0                                                     1UN008
T4       LDL        =0777777777777                                        1UN0090
         STA        CA                    COEF OF A IN CA                 1UN0100
         LDL        =0200000000000000000                                  1UN0110
         STA        S                                                     1UN012
         LDL        =03777000000000000                                    1UN0130
         ARS        36                                                    1UN0140
         LDQ        S                                                     1UN015
         QJP        T9                                                    1UN0160
         SUB        =02000                                                1UN0170
         STA        EA                    EXPONENT OF A IN EA             1UN0180
         SLJ        T10                                                   1UN0190
T9       SUB        =01777                                                1UN0200
         STA        EA                    EXPONENT OF A IN. EA            1UN0210
T10      LDQ        B                     UNPACK B                        1UN0220
         QJP     M  T12                                                   1UN0230
         ENI     2  1                                                     1UN024
         SLJ        T13                                                   1UN0250
T12      LQC        B                                                     1UN0260
         ENI     2  0                                                     1UN027
T13      LDL        =0777777777777                                        1UN0280
         STA        CB                    COEF OF B IN CB                 1UN0290
         LDL        =0200000000000000000                                  1UN0300
         STA        S                                                     1UN031
         LDL        =03777777777777777                                    1UN0320
         ARS        36                                                    1UN0330
```

```
        LDQ       S                                               1UN034
        QJP       T18                                             1UN0350
        SUB       =O2000                                          1UN0360
        STA       EB                EXPONENT OF B IN EB           1UN0370
        SLJ       T19                                             1UN0380
T18     SUB       =O1777                                          1UN0390
        STA       EB                EXPONENT OF B IN EB           1UN0400
T19     ENA    1  0                 PUT UNPACKED NUMBERS IN       1UN0410
        STA       TEMPO             NON-COMPLEMENTARY FORM        1UN0420
+       ENA    2  0                                               1UN043
        ADD       TEMPO                                           1UN0440
        AJP    0  T20                                             1UN0450
        SUB       =O2                                             1UN0460
        AJP    0  T22                                             1UN0470
        ENA    1  0                                               1UN048
        AJP    0  T21                                             1UN0490
        LQC       CB                                              1UN0520
        STQ       CB                                              1UN0530
        SLJ       T22                                             1UN0540
T20     LQC       CB                                              1UN0570
        STQ       CB                                              1UN0580
T21     LQC       CA                                              1UN0610
        STQ       CA                                              1UN0620
T22     ENI    1  **                                              1UN0630
        ENI    2  **                                              1UN0631
T23     SLJ       UNPACK7                                         1UN0632
```



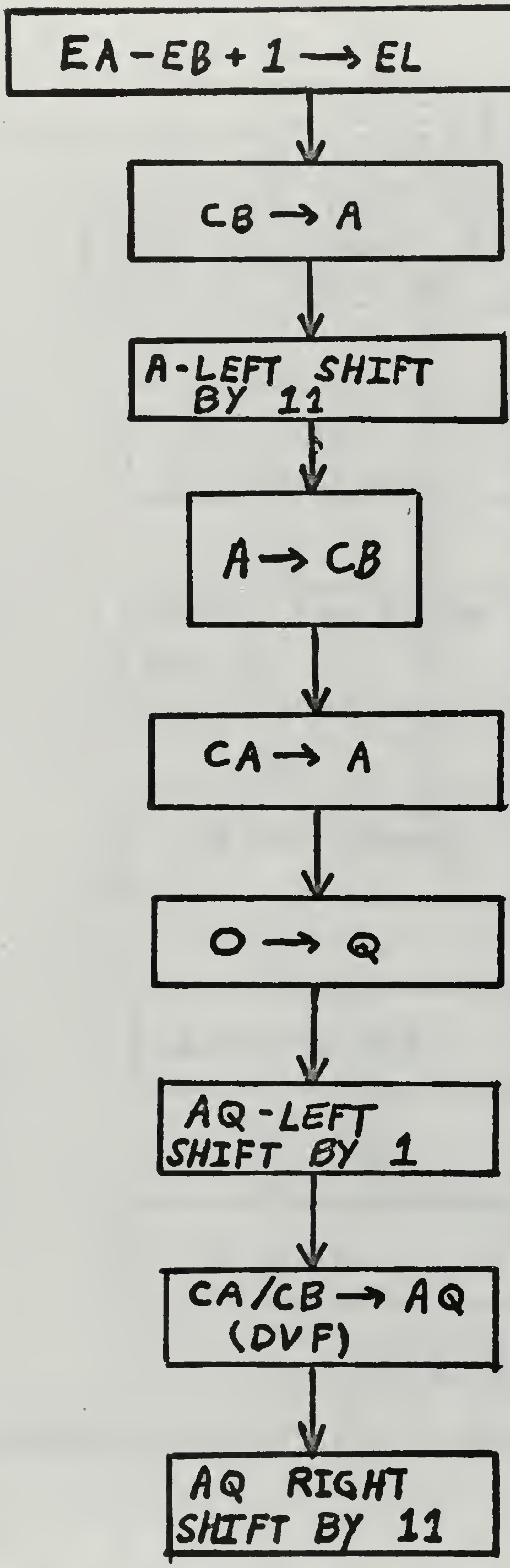
FLOW CHART FOR RNGAD 7

```
         ENTRY     RNGAD7                                                   1AD0000
RNGAD7   SLJ       **                 ADDS TWO UNPACKED NUMBERS             1AD0010
         LDA       EA                                                       1AD0020
         THS       EB                 SEARCH FOR LARGER EXPONENT            1AD0030
         SLJ       P2                                                       1AD0040
P1       LDA       EB                 EB LARGER                             1AD0050
         STA       EL                 STORE EB IN EL                        1AD0060
         SUB       EA                 STORE EXPONENT DIFFERENCE             1AD0070
         SAU       A12                IN A12 UPPER                          1AD0080
         LDA       CB                 STORE CB IN S                         1AD0090
         STA       S                                                        1AD010
         LDA       CA                                                       1AD0110
         SLJ       A12                GO TO A12                             1AD0120
P2       LDA       EA                 EA LARGER                             1AD0130
         STA       EL                 STORE EA IN EL                        1AD0140
         SUB       EB                 EXP DIFFERENCE IN A12 UPPER           1AD0150
         SAU       A12                                                      1AD0160
         LDA       CA                 STORE CA IN S                         1AD0170
         STA       S                                                        1AD018
         LDA       CB                 LOAD CB                               1AD0190
A12      LRS       **                 SHIFT A-RIGHT THE DIFFERENCE          1AD0200
         ADD       S                  IN EXP, AND ADD COEFFICIENTS          1AD0210
         SLJ       RNGAD7                                                   1AD0220
```



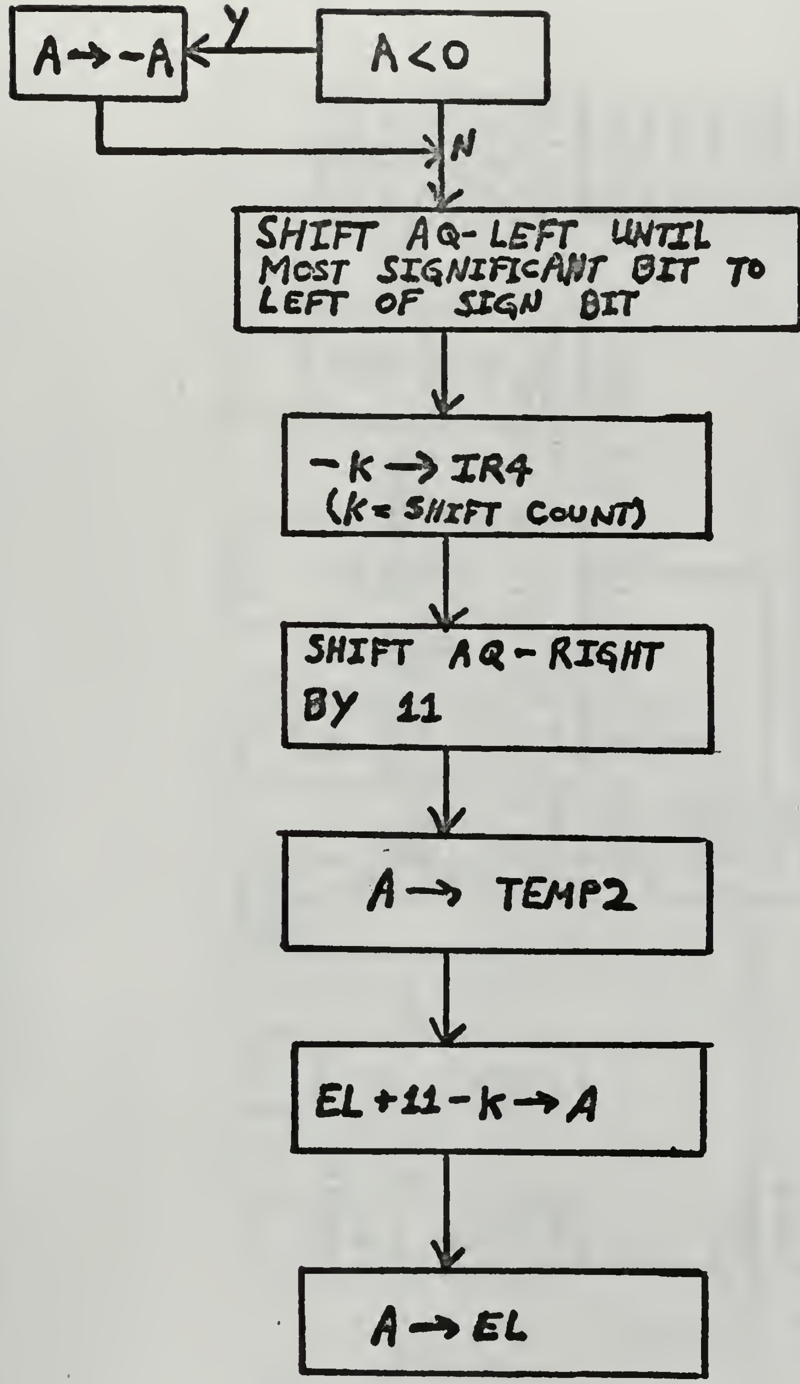
FLOW CHART FOR RNGMU 7

```
         ENTRY   RNGMU7                                     1M0000
RNGMU7   SLJ     **          UN-NORMALIZED MULTIPLY         1M0010
         LDA     EA                                         1M0020
         ADD     EB                                         1M0030
         STA     EL          SUM OF EXPONENTS IN EL         1M0040
         LDA     CA                                         1M0050
         ALS     11          SHIFT CA LEFT 11               1M0060
         STA     CA                                         1M0070
         LDA     CB                                         1M0080
         ALS     11          SHIFT CB LEFT 11               1M0090
         MUF     CA          MULTIPLY COEF FRACTIONALLY     1M0100
         LRS     11          SHIFT A-RIGHT 11               1M0110
         SLJ     RNGMU7                                     1M0120
```



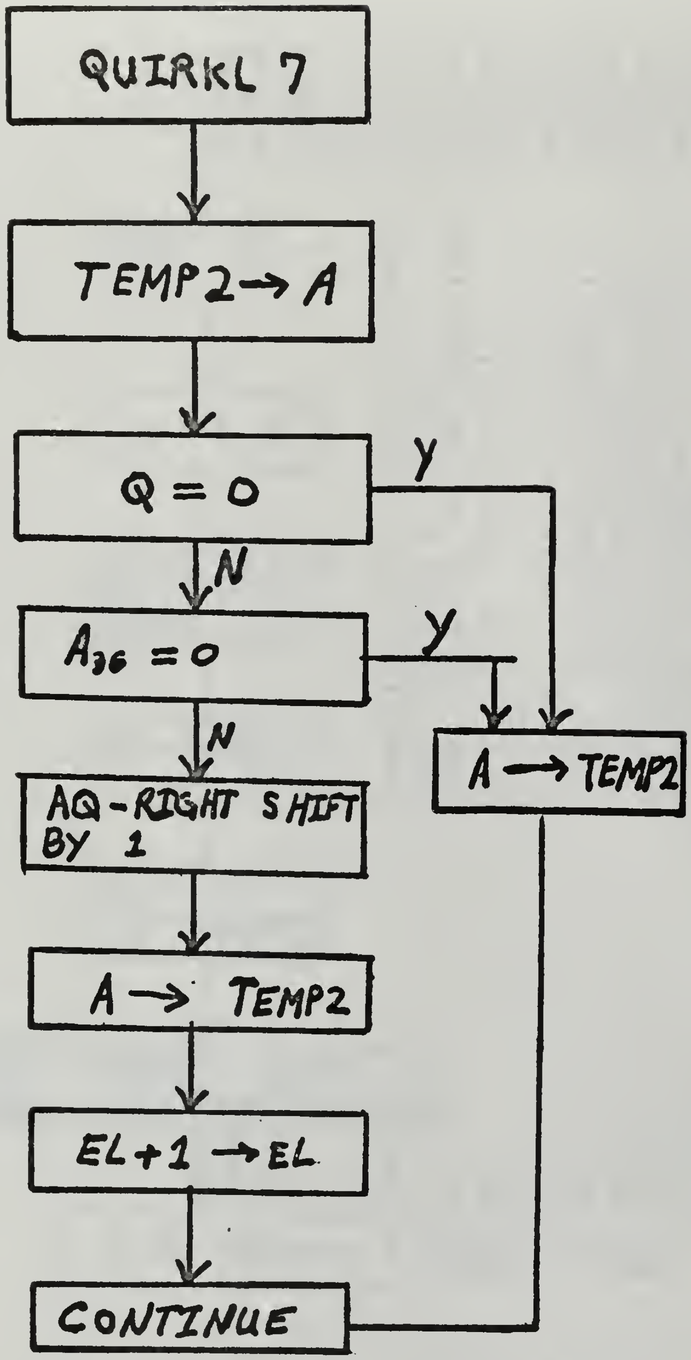
FLOW CHART FOR RNGDI 7

```
        ENTRY    RNGDI7                                        1D0000
RNGDI7  SLJ      **          UN-NORMALIZED DIVISION            1D0010
        LDA      EA                                            1D0020
        SUB      EB                                            1D0030
        INA      +1                                            1D0035
        STA      EL          DIFFERENCE OF EXP +1 IN EL        1D0040
        LDA      CB                                            1D0050
        ALS      11                                            1D0060
        STA      CB                                            1D0070
        LDA      CA          SHIFT CA AND CB EACH RIGHT BY 11  1D0080
        ENQ      0                                             1D009
        LLS      1                                             1D010
        DVF      CB          DIVIDE FRACTIONAL  CA BY CB       1D0110
        LRS      11                                            1D0120
        SLJ      RNGDI7                                        1D0130
```



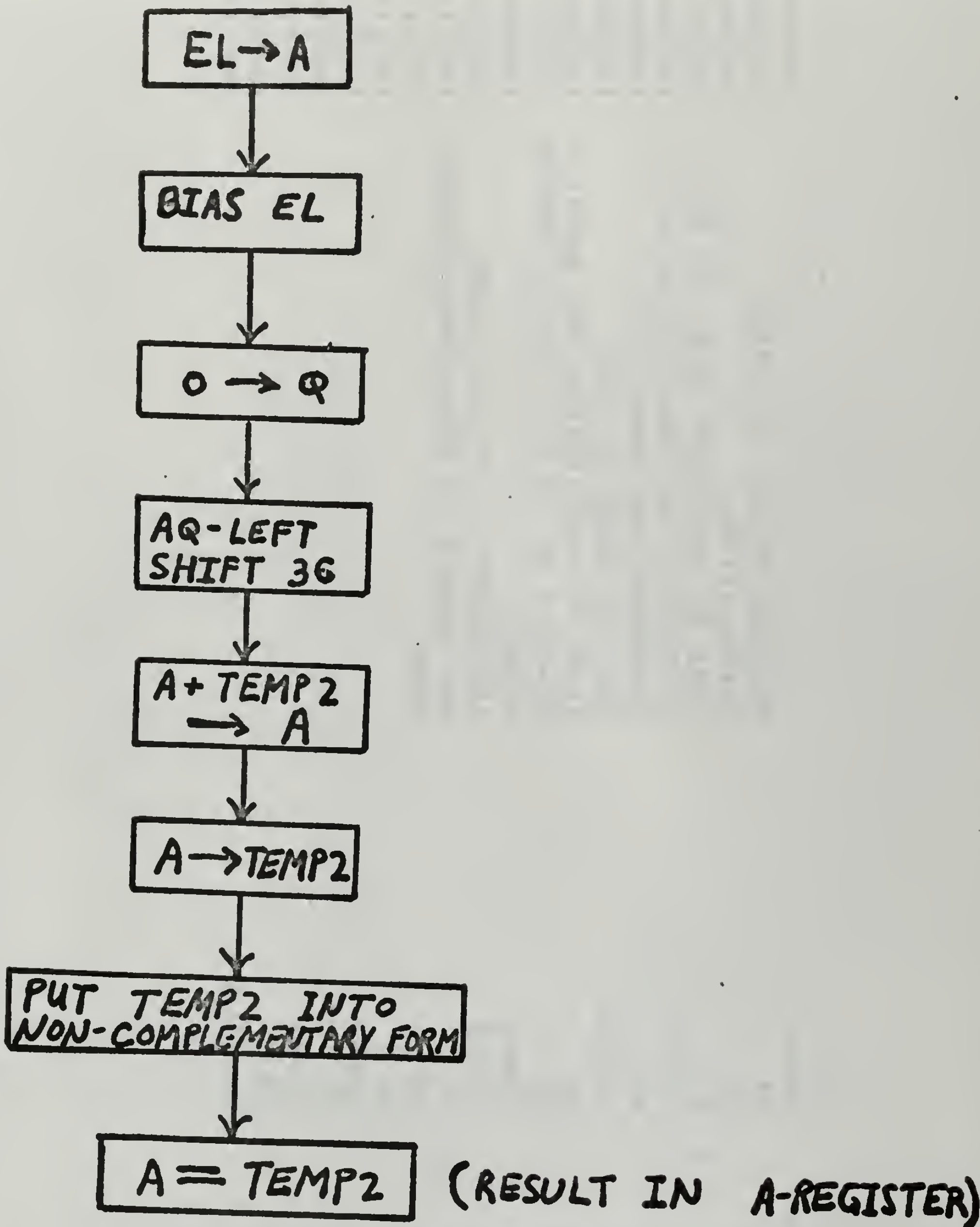
FLOW CHART FOR QUIRKL 7

```
         ENTRY      QUIRKL7                                                 1QL0000
QUIRKL7  SLJ        **                ROUTINE TO TRUNCATE AND               1QL0010
         SIU     4  QL3               NORMALIZE LOWER RANGE NUMBER          1QL0001
         SIU     5  R7                                                      1QL0002
         AJP     2  QL1               IF A-REGISTER IS NEGATIVE,            1QL0020
         ENI     5  0                 COMPLEMENT ACCUMULATOR.               1QL0030
         STA        TEMP4                                                   1QL0040
         LAC        TEMP4                                                   1QL0050
         SLJ        QL2                                                     1QL0060
QL1      ENI     5  1                                                       1QL0070
QL2      ENI     4  47                                                      1QL0080
         SCQ     4  47                SHIFT AQ-REGISTER LEFT UNTIL          1QL0090
         LRS        11                MOST SIGNIFICANT BIT IS TO            1QL0100
         STA        TEMP2             RIGHT OF SIGN BIT. THEN SHIFT         1QL0105
         INI     4  -47               AQ RIGHT BY 11, AND STORE.            1QL0110
         ENA     4  0                                                       1QL012
         INA        11                                                      1QL0130
         ADD        EL                ADJUST EL  DUE TO SHIFT, THAT         1QL0140
         STA        EL                IS, NORMALIZE EXPONENT                1QL0150
QL3      ENI     4  **                                                      1QL0160
         SLJ        QUIRKL7                                                 1QL0170
```



FLOW CHART FOR QUIRKU 7

```
        ENTRY      QUIRKU7                                                 1QU0000
QUIRKU7 SLJ        **                  ROUTINE TO ROUND AND NORMALIZE      1QU0010
        RTJ        QUIRKL7             UPPER RANGE NUMBER.                 1QU0020
+       LDA        TEMP2               CALL ROUTINE FOR NORMALIZING        1QU0030
        QJP     0  QU1                 LOWER RANGE NUMBERS,AND LOAD        1QU0040
        ADD        =O1                 A-REG. WITH COEF. IF Q-REG IS       1QU0050
        LDQ        =O100000000000      ZERO,GO TO QU1. IF NOT,ADD 1        1QU0060
        STL        BIT                 TO COEF.                            1QU0070
        LDQ        BIT                 IF OVERFLOW INTO 37TH BIT,          1QU0080
        QJP     0  QU1                 SHIFT AQ RIGHT BY ONE,AND STORE,    1QU0090
        LRS        1                   THEN INCREASE EL BY ONE.            1QU0100
        STA        TEMP2               OTHERWISE,STORE A-REQ IN TEMP2.     1QU0110
        LDA        EL                                                      1QU0120
        ADD        1                                                       1QU013
        STA        EL                                                      1QU0140
        SLJ        QU2                                                     1QU0150
QU1     STA        TEMP2                                                   1QU0160
QU2     SLJ        QUIRKU7                                                 1QU0170
```



FLOW CHART FOR REPACK 7

```
         ENTRY      REPACK7                                                 1RE0000
REPACK7  SLJ        **                 REPACKS RESULT OF RANGE              1RE0010
         LDA        EL                 CALCULATION.                         1RE0020
         AJP     2  R1                 LOAD EL AND JUMP TO R1 IF POS.       1RE0030
         ADD        =01777             OTHERWISE ADD NEGATIVE BIAS TO       1RE0040
         SLJ        R2                 EXPONENT AND JUMP TO R2              1RE0050
R1       ADD        =02000             ADD POS BIAS TO EXP.                 1RE0060
R2       ENQ        +0                 ZERO IN Q-REGISTER, AND SHIFT        1RE0070
         LLS        36                 AQ-LEFT 36 TO PUT EXP IN PROPER      1RE0080
         ADD        TEMP2              PLACE.                               1RE0090
         STA        TEMP2              ADD COEFFICIENT,AND STORE            1RE0100
         ENA     5  0                  PUT NUMBER IN NON-COMPLEMENTARY      1RE0110
         AJP     0  R3                 FORM                                 1RE0120
         SLJ        R4                                                      1RE0130
R3       LAC        TEMP2                                                   1RE0140
         SLJ        R7                                                      1RE0150
R4       LDA        TEMP2                                                   1RE0160
R7       ENI     5  **                                                      1RE0170
         SLJ        REPACK7                                                 1RE0180
         END
```

APPENDIX II

SUBROUTINES FOR USE WITH QRANGE7 [5]

1. ABS7
2. SQRT7
3. INT7
4. Q0Q06700
5. Q2Q07770

```
         IDENT      ABS7                                          ABS70000
         ENTRY      ABS7                ROUTINE FOR ABSOLUTE VALUE ABS70010
RNG7S777 BLOCK      4                     OF A WHERE A IS A RANGE  ABS70060
         COMMON     ACC(3),B              NUMBER                   ABS70040
ABS7     SLJ        **                                             ABS70050
         LDA        ABS7                                           ABS70060
         ALS        +24                                            ABS70070
         SAU        *+2                                            ABS70080
+        INA        +1                                             ABS70090
         SAL        EXIT+2                                         ABS70100
+        LDA        **                                             ABS70110
         ALS        +24                                            ABS70120
+        SAU        *+1                                            ABS70130
         SIL    1   EXIT                                           ABS70140
+        ENI    1   **                                             ABS70150
         LDA    1   +1                                             ABS70160
         AJP    3   NEG                 A(2) .LT. ZERO             ABS70170
         STA        ACC+1               A(2) .GT. ZERO             ABS70180
         LDA    1   +2                                             ABS70190
         STA        ACC+2                                          ABS70200
         SLJ        EXIT                                           ABS70210
NEG      SCM        =O7777777777777777                             ABS70220
         STA        ACC+2               STORE ABSF(A(2)) IN ACC+2  ABS70230
         LDA    1   +2                                             ABS70240
         AJP    2   POS                 A(3) .GT. ZERO             ABS70250
         SCM        =O7777777777777777  A(3) .LT. ZERO             ABS70260
         STA        ACC+1                                          ABS70270
         SLJ        EXIT                                           ABS70280
POS      THS        ACC+2               A(2) )LT) ZERO .LT. A(3)   ABS70290
         SLJ        *+3                                            ABS70300
+        ENA        0                                              ABS7031
         STA        ACC+1                                          ABS70320
+        SLJ        EXIT                                           ABS70330
+        STA        ACC+2                                          ABS70340
         SLJ        *-2                                            ABS70350
EXIT     LDA    1   +0                                             ABS70360
         ENI    1   **                                             ABS70370
+        AJP    2   *+1                                            ABS70380
         SCM        =O7777777777777777                             ABS70390
+        STA        ACC                                            ABS7400
         SLJ        **                                             ABS70410
```

```
         IDENT       SQRT7                                          SQRT0000
         ENTRY       SQRT7                                          SQRT0010
RNG7S777 BLOCK       4                                              SQRT0020
         COMMON      ACC(3),B                                       SQRT0030
TACC     BSS         3                                              SQRT0040
SQRT7    SLJ         **                                             SQRT0050
         LDA         *                                              SQRT0060
         ALS         +24                                            SQRT0070
         SAU         *+2                                            SQRT0080
+        INA         1                                              SQRT0090
         SAL         EXIT                                           SQRT0100
         LDA         **                                             SQRT0110
         ALS         +24                                            SQRT0120
         SAL         *+1                                            SQRT0130
         SAL         SQRTERR                                        SQRT0140
+        CALL        Q1Q00770                                       SQRT0150
         0           **                                             SQRT0160
         CALL        Q1Q10770                                       SQRT0170
         0           TACC                                           SQRT0180
         LDA         TACC+1                                         SQRT0190
         SIU      1  EXIT                                           SQRT0200
         AJP      M  SQRTERR                                        SQRT0210
         LDA         TACC                                           SQRT0220
         AJP      Z  GO                                             SQRT0225
         LRS         36                                             SQRT0230
+        THS         EXPONCHK                                       SQRT0235
         INA         1                                              SQRT0240
         INA         1777B                                          SQRT0250
         LRS         1                                              SQRT0260
         SAU         ADJUSTEX+1                                     SQRT0270
         ENA         1   B                                          SQRT0280
         LRS         11                                             SQRT0290
         STQ         ERAS                                           SQRT0300
         LDA         A                                              SQRT0310
         FAD         ERAS                                           SQRT0320
         STA         ERAS1                                          SQRT0330
```

```
        LDA      C                                        SQRT0340
        FDV      ERAS1                                    SQRT0350
        FAD      A                                        SQRT0360
        STA      ERAS1                                    SQRT0370
        LDA      ERAS                                     SQRT0380
        FDV      ERAS1                                    SQRT0390
        FAD      ERAS1                                    SQRT0400
        SUB      DIV2                                     SQRT0410
        STA      ERAS1                                    SQRT0420
        LDA      ERAS                                     SQRT0430
        FDV      ERAS1                                    SQRT0440
        FAD      ERAS1                                    SQRT0450
        LRS      36                                       SQRT0460
        RTJ      ADJUSTEX                                 SQRT0470
GO      STA      TACC                                     SQRT0480
        ENI    1 1                                        SQRT0490
LOOP    LDA    1 TACC+1                                   SQRT0500
        AJP    Z GOE                                      SQRT0510
        LRS      36                                       SQRT0520
+       THS      EXPONCHK                                 SQRT0530
        INA      1                                        SQRT0540
        INA      1777B                                    SQRT0550
        LRS      1                                        SQRT0560
        SAU      ADJUSTEX+1                               SQRT0570
        ENA      1   B                                    SQRT0580
        LRS      11                                       SQRT0590
        STQ      ERAS                                     SQRT0600
        LDA      ERAS                                     SQRT0610
        STA      ERAS+1                                   SQRT0620
        ENA      0                                        SQRT0630
        STA      B                                        SQRT0640
        LDA      ERAS                                     SQRT0650
        CALL     UNPACK7                                  SQRT0660
+       CALL     RNGAD7                                   SQRT0670
+       CALL     QUIRKU7                                  SQRT0680
+       CALL     REPACK7                                  SQRT0690
```

```
+       STA     ERAS+2                                      SQRT0700
+       CALL    Q1Q00770                                    SQRT0710
        0       A                                           SQRT0720
        CALL    Q1Q02770                                    SQRT0730
        0       ERAS                                        SQRT0740
        CALL    Q1Q10770                                    SQRT0750
        0       ERAS1                                       SQRT0760
        CALL    Q1Q00770                                    SQRT0770
        0       C                                           SQRT0780
        CALL    Q1Q05770                                    SQRT0790
        0       ERAS1                                       SQRT0880
        CALL    Q1Q02770                                    SQRT0810
        0       A                                           SQRT0820
        CALL    Q1Q10770                                    SQRT0830
        0       ERAS1                                       SQRT0840
        CALL    Q1Q00770                                    SQRT0850
        0       ERAS                                        SQRT0860
        CALL    Q1Q05770                                    SQRT0870
        0       ERAS1                                       SQRT0880
        CALL    Q1Q02770                                    SQRT0890
        0       ERAS1                                       SQRT0900
        LDA     ACC                                         SQRT0910
        SUB     DIV2                                        SQRT0920
        STA     ERAS1                                       SQRT0930
        LDA     ACC+1                                       SQRT0940
        SUB     DIV2                                        SQRT0950
        STA     ERAS1+1                                     SQRT0960
        LDA     ACC+2                                       SQRT0970
        SUB     DIV2                                        SQRT0980
        STA     ERAS1+2                                     SQRT0990
+       CALL    Q1Q00770                                    SQRT1000
        0       ERAS                                        SQRT1005
        CALL    Q1Q05770                                    SQRT1010
        0       ERAS1                                       SQRT1020
        CALL    Q1Q02770                                    SQRT1030
        0       ERAS1                                       SQRT1040
```

```
          LDA     1  ACC+1                                    SQRT1050
          LRS        36                                       SQRT1060
          RTJ        ADJUSTEX                                 SQRT1070
GOE       STA     1  TACC+1                                   SQRT1080
          IJP     1  LOOP                                     SQRT1090
          CALL       Q1Q00770                                 SQRT1100
          0          TACC                                     SQRT1110
EXIT      ENI     1  **                                       SQRT1120
          SLJ        **                                       SQRT1130
ADJUSTEX  SLJ        **                                       SQRT1140
          INA        -4000B                                   SQRT1150
          INA        **                                       SQRT1160
+         AJP     Z  NORM                                     SQRT1170
          INA        1777B                                    SQRT1180
          LLS        36                                       SQRT1190
          SLJ        ADJUSTEX                                 SQRT1200
NORM      INA        1                                        SQRT1210
          SLJ        ADJUSTEX+2                               SQRT1220
SQRTERR   CALL       ERROR777                                 SQRT1230
+         0          SQRT7                                    SQRT1240
          0          **                                       SQRT1250
          BCD        1 SQRT7                                  SQRT1260
+         SLJ        EXIT                                     SQRT1270
ERAS      BSS        3                                        SQRT1280
ERAS1     BSS        3                                        SQRT1290
DIV2      OCT        1000000000000                            SQRT1300
EXPONCHK  OCT        1777                                     SQRT1310
A         OCT        2002613436725763                         SQRT1320
          OCT        2  2613436725760                         SQRT1330
          OCT        2  2613436725764                         SQRT1340
C         OCT        5773356313766036                         SQRT1350
          OCT        5773356313766033                         SQRT1360
          OCT        5773356313766037                         SQRT1370
          END                                                 SQRT1380
```

```
         IDENT      INT7                                               INT70010
         ENTRY      INT7                                               INT70020
RNG7S777 BLOCK      4                                                  INT70030
         COMMON     ACC(3),B                                           INT70040
INT7     SLJ        **                 TRUNCATE RANGE7 NUMBER AND      INT70050
         SIU      1 EXIT                 RETURN AS A RANGE7 NUMBER     INT70060
         LIU      1 *-1                                                INT70070
         SIU      1 *+2                                                INT70080
         INI      1 1                                                  INT70090
         SIL      1 EXIT                                               INT70100
         LIU      1 **                                                 INT70110
         LDA      1 0                                                  INT70120
         CALL       INTF                                               INT70130
+        STA        ACC                                                INT70140
         INI      1 1                                                  INT70150
         LDA      1 0                                                  INT70160
         CALL       INTF                                               INT70170
+        STA        ACC+1                                              INT70180
         INI      1 1                                                  INT70190
         LDA      1 0                                                  INT70200
         CALL       INTF                                               INT70210
+        STA        ACC+2                                              INT70220
         LDA        ACC                                                INT70230
EXIT     ENI      1 **                                                 INT70240
         SLJ        **                                                 INT70250
         END                                                           INT70260
```

```
         IDENT     Q0Q06700
RNG7S777 BLOCK     4
         COMMON    ACC(3),B
Q        BSS       4
         ENTRY     Q0Q06700                 COMPLEMENT RANGE ACCUMULATOR    Q3000010
Q0Q06700 SLJ       **                                                       Q3000020
         LDA       ACC+2                                                    Q3000030
         STA       Q+3                                                      Q3000040
         LAC       ACC+1                                                    Q3000050
         STA       ACC+2                                                    Q3000060
         LAC       Q+3                                                      Q3000070
         STA       ACC+1                                                    Q3000080
         LAC       ACC                                                      Q3000090
         STA       ACC                                                      Q3000100
         SLJ       Q0Q06700                                                 Q3000110
         END
```

```
         IDENT     Q2Q07770                                  Q2770000
         ENTRY     Q2Q07770     ROUTINE FOR X**I, WHERE      Q2770010
RNG7S777 BLOCK     4            X IS RANGE AND I IS INTEGER  Q2770030
         COMMON    ACC(3),B                                  Q2770040
ONE      DEC       1.                                        Q2770050
         DEC       1.                                        Q2770060
         DEC       1.                                        Q2770070
X        BSS       3                                         Q2770080
XTO2N    BSS       3                                         Q2770090
I        BSS       1                                         Q2770100
ISAVE    BSS       1                                         Q2770110
Q2Q07770 SLJ       **                                        Q2770120
         SIU     1 EXIT                                      Q2770130
         LDA       *-1                                       Q2770140
         ALS       +24                                       Q2770150
         INA       -1                                        Q2770160
         SAU       START                                     Q2770170
         INA       -1                                        Q2770180
         SAU       START+1                                   Q2770190
START    ENA     7 **           ADDRESS OF I                 Q2770200
         SAU       NOW                                       Q2770210
         ENA     7 **           ADDRESS OF X                 Q2770220
         SAL       *+1                                       Q2770230
+        CALL      Q1Q00770                                  Q2770240
         0         **                                        Q2770250
NOW      LDQ       **           PUT I IN Q-REGISTER          Q2770260
         QJP     P PROC         ERROR EXIT IF I IS NEGATIVE  Q2770270
         LDA       ACC+1        AND ZERO IS IN RANGE OF X.   Q2770280
         AJP     P PROC                                      Q2770290
         LDA       ACC+2                                     Q2770300
         AJP     M PROC                                      Q2770310
ERREXT   CALL      ERROR777     NOTE.  0**0 IS EQUAL TO ONE  Q2770320
+        0         Q2Q07770                                  Q2770330
         0         ACC                                       Q2770340
         BCD       1 X**I                                    Q2770350
```

```
PROC     STQ        ISAVE                    Q2770360
+        QJP     P  *+1                      Q2770370
         LQC        ISAVE                    Q2770380
         STQ        I                        Q2770390
+        CALL       Q1Q10770                 Q2770400
         0          XTO2N                    Q2770410
         LDA        I                        Q2770420
         LRS        +1                       Q2770430
         STA        I                        Q2770440
+        CALL       Q1Q00770                 Q2770450
         0          ONE                      Q2770460
         QJP     P  *+2                      Q2770470
+        CALL       Q1Q00770                 Q2770480
         0          XTO2N                    Q2770490
         ENI     1  0                        Q2770500
+        CALL       Q1Q10770                 Q2770510
         0          X                        Q2770520
LOOPB    LDA        I                        Q2770530
         AJP     Z  OUT                      Q2770540
LOOPB1   LRS        +1                       Q2770550
         QJP     M  LOOPC                    Q2770560
         INI     1  +1                       Q2770570
         SLJ        LOOPB1                   Q2770580
LOOPC    STA        2                        Q2770590
+        CALL       Q1Q00770                 Q2770600
         0          XTO2N                    Q2770610
LOOPC1   CALL       Q1Q04770                 Q2770620
         0          XTO2N                    Q2770630
         CALL       Q1Q10770                 Q2770640
         0          XTO2N                    Q2770650
         IJP     1  LOOPC1                   Q2770660
+        CALL       Q1Q04770                 Q2770670
         0          X                        Q2770680
         CALL       Q1Q10770                 Q2770690
         0          X                        Q2770700
         SLJ        LOOPB                    Q2770710
```

```
OUT      SSK        ISAVE                                        Q2770720
         SLJ        RELOAD                                       Q2770730
+        CALL       Q1Q00770                                     Q2770740
         0          ONE                                          Q2770750
         CALL       Q1Q05770                                     Q2770760
         0          X                                            Q2770770
EXIT     ENI     1  **                                           Q2770780
         SLJ        Q2Q07770                                     Q2770790
RELOAD   CALL       Q1Q00770                                     Q2770800
         0          X                                            Q2770810
         SLJ        EXIT                                         Q2770820
         END                                                     Q2770830
```

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2.	Library U. S. Naval Postgraduate School Monterey, California 93940	2
3.	Professor Rex H. Shudde 167 Arriba Way, Route #1 Carmel, California 93921	1
4.	Lieutenant Victor Joseph Monte Leon 1052 Third Street Monterey, California 93940	1

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* U. S. Naval Postgraduate School Monterey, California	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

AUTOMATIC ERROR ANALYSIS IN FINITE DIGITAL COMPUTATIONS, USING RANGE ARITHMETIC

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Master's Thesis

5. AUTHOR(S) *(Last name, first name, initial)*

Monte Leon, Victor J., Lieutenant, United States Navy

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
May, 1966	81	5

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	

10. AVAILABILITY/LIMITATION NOTICES

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY

13. ABSTRACT

The nature of generated machine error in finite digital calculations is discussed. The arithmetic of range numbers is developed, and examples are given demonstrating the use of range arithmetic as a tool for automatic error analysis. A computer program is developed, utilizing the TYPE OTHER feature of FORTRAN-63 in conjunction with the CDC-1604 digital computer, which enables the user to perform automatic error analysis during computation, and a number of programs are presented using this feature.

DD FORM 1 JAN 64 1473

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Automatic Error Analysis						
Range Arithmetic						
Interval Arithmetic						
Digital Computation						
Finite Precision Accuracy						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8c, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
_______________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
_______________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
_______________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.

DD FORM 1 JAN 64 1473 (BACK)

[redacted]